



## COURSE INFORMATION FORM

<b>Faculty</b>	<b>Faculty of Engineering</b>		
<b>Program</b>	<b>B.Sc. in Civil Engineering</b>	<b>Elective</b>	
	<b>B.Sc. in Computer Engineering</b>	<b>Required</b>	
	<b>B.Sc. in Electrical-Electronics Engineering</b>	<b>Required</b>	
	<b>B.Sc. in Industrial Engineering</b>	<b>Elective</b>	
	<b>B.Sc. in Mechanical Engineering</b>	<b>Elective</b>	

<b>Course Code</b>	COMP 110			
<b>Course Title in English</b>	Object Oriented Programming (Java)			
<b>Course Title in Turkish</b>	Nesne Yönelimli Programlama (Java)			
<b>Language of Instruction</b>	English			
<b>Type of Course</b>	Flipped Classroom/Lecture/Exercise			
<b>Level of Course</b>	Undergraduate			
<b>Course Category (by % of Content)</b>	Basic Science 10	Basic Engineering 50	Engineering Design 30	General Education 10
<b>Semester Offered</b>	Fall			
<b>Contact Hours per Week</b>	Lecture: 3 hours	Recitation: -	Lab: 2 hours	Other:-
<b>Estimated Student Workload</b>	150 hours per semester			
<b>Number of Credits</b>	6 ECTS			
<b>Grading Mode</b>	Standard Letter Grade			
<b>Pre-requisites</b>	COMP 109			
<b>Expected Prior Knowledge</b>	Basic programming knowledge			
<b>Co-requisites</b>	None			
<b>Registration Restrictions</b>	Only Undergraduate Students			
<b>Overall Educational Objective</b>	To learn fundamentals of object-oriented programming and how to design and implement object-oriented computer algorithms to solve basic engineering problems in Java programming language.			
<b>Course Description</b>	This course covers the fundamentals of object-oriented programming approach such as objects, classes, inheritance, polymorphism, dynamic binding, and application of these concepts using Java programming language.			
<b>Course Description in Turkish</b>	Nesne yönelimli programlama dersi nesne, sınıf, kalıtım gibi nesne yönelimli programlama kavramları ve bunların Java programlama dili kullanılarak uygulanmasını içermektedir.			
<b>Course Learning Outcomes and Competences</b>	Upon successful completion of the course, the learner is expected to: 1. understand basic object-oriented concepts; objects, classes, inheritance, polymorphism; 2. design object-oriented algorithms; 3. implement object-oriented computer programs to solve simple engineering problems; 4. use Java programming language.			

<b>Relationship of the Course with the Student Outcomes</b>	<b>Level</b>	<b>Learning Outcome(s)</b>	<b>Assessed by</b>
<b>Program Outcomes</b>	N=None S=Supportive H=High		Exam, Project, HW, Experiment, Presentation, etc.
(a) an ability to apply knowledge of mathematics, science, and engineering	S	1	Exams
(b) an ability to design and conduct experiments, as well as to analyze			

and interpret data			
(b)-1. an ability to design/develop an experiment by identifying required assumptions, constraints, data collection methods and models			
(b)-2. Implement experimental procedures to conduct an experiment and use engineering judgment to draw conclusions			
(c) an ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability	H	1,2,3,4	Exams, Assignments
(d) an ability to function on multidisciplinary teams			
(d)-1. Function effectively on an intradisciplinary team			
(d)-2. Function effectively on a multidisciplinary team			
(e) an ability to identify, formulate, and solve engineering problems	H	1,2,3,4	Exams, Assignments
(f) an understanding of professional and ethical responsibility	S	3	Assignments
(g) an ability to communicate effectively			
(g)-1. Communicate effectively with well-organized written documents	S	3	Assignments
(g)-2. Communicate effectively verbally with a range of audiences			
(h) the broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context			
(i) a recognition of the need for, and an ability to engage in life-long			
(j) a knowledge of contemporary issues			
(k) an ability to use the techniques, skills, and modern engineering tools necessary for engineering practice	H	3,4	Assignments
<b>Prepared by and Date</b>	Berk Gökberk, 1 October 2016		
<b>Semester</b>	Spring 2017-2018		
<b>Name of Instructor</b>	Berk Gökberk		
<b>Course Contents</b>	Week	Topic	
	1.	Introduction to Object Oriented Programming Concepts	
	2.	Control structures, Loops	
	3.	Arrays and Array Lists	
	4.	Methods	
	5.	Classes Part 1	
	6.	Classes Part 2	
	7.	Object Oriented Design Part 1 (Class Design)	
	8.	Inheritance Part 1	
	9.	Inheritance Part 2	
	10.	Polymorphism Part 1	
	11.	Polymorphism Part 2	
	12.	Object Oriented Design Part 2 (Class Hierarchy Design)	
	13.	Exception Handling	
	14.	Object Oriented Design Part 3 (Advanced Design)	
	15.	Final Examination Period	
	16.	Final Examination Period	
<b>Required/Recommended Readings</b>	Intro. to Java Programming: Comprehensive Ed. (10th Ed., Pearson, 2014), Daniel Liang.		

<b>Teaching Methods</b>	Flipped classroom. Students work individually for assignments.
<b>Homework and Projects</b>	Assignments
<b>Laboratory Work</b>	Laboratory study
<b>Computer Use</b>	Required
<b>Other Activities</b>	-
<b>Assessment Methods</b>	2 Midterm Exams (50% total), Final Exam (35%), Assignments (15%)
<b>Course Administration</b>	Instructor's office: 5th floor Phone number: 0 212 395 37 45 Office hours: After the lecture hours. E-mail address: gokberkb@mef.edu.tr Rules for attendance: No attendance required. Statement on plagiarism: YÖK Regulations <a href="http://3fcampus.mef.edu.tr/uploads/cms/webadmin.mef.edu.tr/4833_2.pdf">http://3fcampus.mef.edu.tr/uploads/cms/webadmin.mef.edu.tr/4833_2.pdf</a>

ECTS Student Workload Estimation	Activity	No/Weeks	Hours			Calculation	Explanation
		No/Weeks per Semester (A)	Preparing for the Activity (B)	Spent in the Activity Itself (C)	Activity Requirements (D)		
	Lecture/Flipped Classroom	14	2	3		70	A*(B+C+D)
	Quizzes					0	
	Lab	14		2		28	
	Midterm(s)	2	10	2	0	24	A*(B+C+D)
	Assingments	4	1	1	0	8	A*(B+C+D)
	Final Examination	1	18	2	0	20	A*(B+C+D)
	Total Workload					150	
	Total Workload/25					6	

### PROGRAM CRITERIA

1. Breadth in computer engineering practice, analysis and design with 18 required courses, and depth in one or more fields with 4 electives.
2. Knowledge of mathematics, including differential and integral calculus, basic sciences, computer science, and engineering sciences that is necessary for analysis and design of complex electrical and electronic devices, software, and systems containing hardware and software components.
3. Knowledge of probability and statistics, including application to computer engineering; knowledge of advanced mathematics, including differential equations, linear algebra, complex variables, and discrete mathematics.

Note: For program-specific courses ABET Program Criteria of the related engineering program will be put here as before.