Course Description Form	
Course Code and Name	CENG491 SYSTEM PROGRAMMING (TECH.ELECT.)
Course Semester	7
Catalog Content	Loader, Connector, Micro programming, single and double-pass symbolic converters, Design and implementation of a variety of system software, The relationship between machine architecture and system software, Windows, Unix operating systems, the introduction
Textbook	The Linux Programming Interface: A Linux and UNIX System Programming Handbook, Michael Kerrisk, 2010.
Supplementary Textbooks	Advanced Programming in the Unix Environment, 3rd Edition, by Richard Stevens and Steven A Rago, Addison-Wesley, 2013 Understanding UNIX/LINUX Programming: A Guide to Theory and Practice, by Bruce Molay, Prentice Hall, 2002.
Credit	6
Prerequisites of the Course **(** *Attendance Requirements***)**	There is no prerequisite or co-requisite for this course.
Type of the Course	Elective
Instruction Language	English
Course Objectives	This course aims to give current information about: Symbolic programming elements. Source and object program, re-entered the program, run the program again. Addressing techniques, The concept of procedure
Course Learning Outcomes	1. Grasping the following topics and concepts: Symbolic programming elements. Source and object program, re-entered the program, run the program again. Addressing techniques, The concept of procedure
Instruction Methods	The mode of delivery of this course is face to face.

<table>
<tr><td>Weekly Schedule</td><td>1. Week: Symbolic programming elements
2. Week: Source and object program
3. Week: Re-enter the program
4. Week: Re-run the program
5. Week: Addressing techniques, the concept of Procedure
6. Week: Parameter communication techniques
7. Week: Operating string-symbolic relationship between the program, the programming techniques
8. Week: Loader, Connector, Micro programming, single and double-pass symbolic converters
9. Week: Design and implementation of a variety of system software
10. Week: The relationship between machine architecture and system software
11. Week: Windows, Unix operating systems, the introduction
12. Week: Assembly languages
13. Week: Machine-dependent assembly, machine independent assembly
14. Week: Program blocks. Assembler design, MASM and SPARC structures.</td></tr>
<tr><td>Teaching and Learning Methods

(These are examples. Please fill which activities you use in the course)</td><td>Weekly theoretical course hours: 3
Reading Activities
Internet browsing, library work
Preparation of Midterm and Midterm Exam
Final Exam and Preparation for Final Exam</td></tr>
<tr><td>Assessment Criteria</td><td>
<table>
<tr><th></th><th>Numbers</th><th>Total Weighting (%)</th></tr>
<tr><td>Midterm Exams</td><td>1</td><td>30</td></tr>
<tr><td>Assignment</td><td>5</td><td>30</td></tr>
<tr><td>Application</td><td></td><td></td></tr>
<tr><td>Projects</td><td></td><td></td></tr>
<tr><td>Practice</td><td></td><td></td></tr>
<tr><td>Quiz</td><td></td><td></td></tr>
<tr><td>Percent of In-term Studies (%)</td><td></td><td>60</td></tr>
<tr><td>Percentage of Final Exam to Total Score (%)</td><td></td><td>40</td></tr>
<tr><td>Attendance</td><td></td><td></td></tr>
</table>
</td></tr>
</table>

Workload

Activity	Total Number of Weeks	Duration (weekly hour)	Total Period Work Load
Weekly Theoretical Course Hours	14	3	42
Weekly Tutorial Hours			
Reading Tasks	12	4	48
Studies	10	3	30
Material Design and Implementation			
Report Preparing			
Preparing a Presentation			
Presentations			
Midterm Exam and Preparation for Midterm Exam	1	15	15
Final Exam and Preparation for Final Exam	1	15	15
Other (should be emphasized)			
Total Workload			150
Total Workload / 25			6
Course Credit (ECTS)			6

Contribution Level Between Course Learning Outcomes and Program Outcomes

No	Program Outcomes	1	2	3	4	5
1	Sufficient knowledge on mathematics, science and computer engineering; ability to apply theoretical and practical knowledge in these areas to model and solve engineering problems				X	
2	Ability to identify, define, formulate and solve complex engineering problems; ability to choose and apply appropriate analysis and modelling methods for these purposes					X
3	Ability to design a complex system, process, device, software, algorithm, or product under realistic constraints and circumstances to meet certain requirements; ability to apply modern design techniques for this purpose			X		
4	Ability to choose, develop and use modern techniques and tools necessary for engineering applications; ability to effectively use computing technologies					X
5	Ability to design and implement systems or experiments to solve engineering problems, collect and interpret data to evaluate and analyze the results of solutions					X
6	Ability to work effectively in intradisciplinary and interdisciplinary teams or individually				X	
7	Ability to efficiently prepare, evaluate and interpret reports				X	
8	Ability to make presentations and conduct effective verbal and written communication in Turkish and English			X		
9	Awareness of the necessity of lifelong learning; ability to access information, follow scientific and technological developments; ability to perpetually renew oneself					X
10	Awareness of professional and ethical responsibility, ability to act in accordance with ethical principles				X	

	11	Ability to apply knowledge on project management, risk management and change management					X
	12	Awareness of entrepreneurship and innovation, ability to design and build sustainable systems			X		
	13	Ability to devise local and global solutions to contemporary issues considering the effects of engineering applications on health, environment and security		X			
	14	Awareness of the legal consequences of engineering solutions	X				
	15	Ability to apply knowledge on software development process and documentation rules		X			
	16	Knowledge on standards used in engineering applications				X	
	17	Awareness of occupational health and security, information security and privacy		X			
The Course's Lecturer(s) and Contact Information	Computer Engineering Department Chair bmbb@gazi.edu.tr						