Ilia State University
Faculty of Business, Technology and Education
Bachelor's Level
Course Syllabus

1.	**COURSE TITLE**	**T129 System Programming**
2.	**COURSE DURATION**	1 Semester
3.	**ECTS CREDITS**	6 Credits
4.	**DISTRIBUTION OF HOURS**	**Contact Hours: 77 hours** Lectures - 29 hours Practical Work – 34 hours Recitation - 10 hours Mid-term exam - 2 hours Final Exam - 2 hours **Independent Work: 73 hours**
5.	**INSTRUCTOR**	**Amiran Malania, Invited Lecturer** E-mail: amiran.malania@iliauni.edu.ge ***Note:*** *Office hours will be scheduled by the beginning of the semester and students will be notified through the Argus system.*
6.	**PREREQUISITES**	Object Oriented Programming
7.	**INSTRUCTION METHODS**	• Lecture; • Practical Work; • Laboratory Work; • Project.
8.	**AIMS OF THE COURSE**	This course aims to provide students with a deep understanding of computer systems from a programmer's perspective. By studying key underlying principles of different aspects of computer systems such as: machine organization, information processing, memory model, virtualization and models of concurrent programming, successful graduates of this course will be able to write robust, safe and reliable programs.
9.	**MAJOR TOPICS**	• Representing and manipulating information • Machine-level representations of programs • Optimizing program performance • The memory hierarchy • Linking • Virtual memory • System-level I/O • Network programming • Concurrent programming

<table>
<tr>
<td>10.</td>
<td>COURSE OBJECTIVES: LEARNING OUTCOMES AND COMPETENCES</td>
<td>
<u>Upon successful completion of this course, students will acquire the following competencies:</u>

Course Learning Outcomes:
1. Student understands the basic concepts of system calls, error returns, the I/O operations and behaviors available via the system calls, and the use of available resources to uncover the details of how the system calls must be invoked and how they can be used.
2. Student develops small application programs using system calls for interprocess communications.
3. Student analyzes system communications interactions for synchronous and asynchronous communications.
4. Student uses Linux commands and tools to edit, compile, run, and debug programs, and to view and manage processes and interprocess communications resources.

The following table shows the relationship between course learning outcomes and Program Learning Outcomes (PLOs):
<table>
<tr><td>PLO#</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td></tr>
<tr><td>Correlation</td><td>X</td><td>X</td><td></td><td></td><td></td><td>X</td><td></td></tr>
</table>
</td>
</tr>
<tr>
<td>11.</td>
<td>EVALUATION AND GRADING</td>
<td>
Assessment is based on a 100-point scale. Points in the assessment system are distributed in the following way:
(A) 91 - 100 Excellent
(B) 81 - 90 Very good
(C) 71 - 80 Good
(D) 61 - 70 Satisfactory
(E) 51 - 60 Sufficient
(FX) 41 – 50 Unsatisfactory, a student needs more efforts to pass an examination and is given an extra chance to pass an additional examination through independent work
(F) 0 – 40 Failure, student's effort is not sufficient and s/he has to retake a course

Assessment Components and Criteria:
Assessment components include midterm and final exams, lab work assignments.
</td>
</tr>
</table>

The contribution of each component to the final score is as follows:

Assessment	Thresholds	Contributing to Final Grade
Midterm Exam	**10%**	20%
Lab Work Assignments	**25%**	50%
Final Exam	**15%**	**30%**
Total		**100%**

FINAL and MidTerm EXAM IS MANDATORY FOR EVERY STUDENT!

1. **Midterm Exam**

*Throughout the course, there is **one midterm exam** graded with a maximum of **71 points*** contributing **20%** of final grade. *The mid-term includes programming **problems**, **multiple-choice*** and ***open*** questions of varying difficulty.

Assessment criteria for Multiple-choice Questions:

Correct answer	Wrong answer
1	0

Assessment criteria for Open Questions:

	1	2	3
Understanding of the concept	The Student **vaguely** explains a concept, **but** cannot correctly answer qualitative questions, (e.g. wrong theoretical facts are considered a supporting argumentation of the answer).	Student **correctly** explains a concept, **but** cannot correctly answer qualitative questions, (e.g. theoretical statements are correct, however not sufficient for argumentation of answer).	Student **correctly** and briefly explains a concept **and** correctly answers qualitative questions, (e.g. all theoretical facts are provided correctly for argumentation of the correct answer on the question).

Assessment criteria for Problems:

	1	2	3	4	5

Student correctly solves assigned problem	The answer is technically correct, but the overall logic of arrival to the solution is missing.	Overall logic of arrival to the solution is partially present, but has one or more major or conceptual mistakes.	Overall logic of arrival to the solution is partially present and correct.	Has at most two minor mistakes and the overall logic of arrival to the solution is correct.	The solution is correct, the overall logic of arrival to the solution is correct.

2. Lab Work Assignments

Labs	**Weight**
Data Lab	6 %
Bomb Lab	7%
Attack Lab	0%
Cache Lab	9%
Shell Lab	8%
Malloc Lab	12%
Proxy Lab	8%

Completion of 7 ***lab work assignments will be evaluated with varying points depending on the lab.*** *Overall labs - which will be submitted as a small project -* ***contribute 50% to the final grade****. Each lab has individual weight assigned to it and it varies from one lab to another. The instructor will provide students with the descriptions of the assignments and guidelines before the lab. Each lab will be assessed by checking:*

Meeting the requirements of the assignments

0%-10%	**10%-30%**	**30%-60%**	**60%-90%**	**90%-100%**
Poor/Not Submitted	With Major Mistake	Average	With Minor Mistakes	Good

3. Final Exam

The final exam is a written exam based on all the materials covered during the semester. The Exam paper consists of programming ***problems, multiple-choice*** and ***open*** questions of varying difficulty . *The exam is graded with a* ***maximum of 100 points*** *and* ***contributes 30%*** *to final grade.*

Assessment criteria for Multiple-choice Questions:

Correct answer	Wrong answer
1	0

Assessment criteria for Open Questions:

	1	2	3
Understanding of the concept	The Student **vaguely** explains a concept, **but** cannot correctly answer qualitative questions, (e.g. wrong theoretical facts are considered a supporting argumentation of the answer).	Student **correctly** explains a concept, **but** cannot correctly answer qualitative questions, (e.g. theoretical statements are correct, however not sufficient for argumentation of answer).	Student **correctly** and briefly explains a concept **and** correctly answers qualitative questions, (e.g. all theoretical facts are provided correctly for argumentation of the correct answer on the question).

Assessment criteria for Problems:

	1	2	3	4	5
Student correctly solves assigned problem	The answer is technically correct, but the overall logic of arrival to the solution is missing.	Overall logic of arrival to the solution is partially present, but has one or more major or conceptual mistakes.	Overall logic of arrival to the solution is partially present and correct.	Has at most two minor mistakes and the overall logic of arrival to the solution is correct.	The solution is correct, the overall logic of arrival to the solution is correct.

12. TEXTS AND ADDITIONAL RESOURCES

Required reading:

- Randal E. Bryant, David R. O'Hallaron Computer Systems: A Programmer's Perspective, 3/E(CS:APP3e), 2015, ISBN-13: 978-0134092669

Recommended:

<table>
<tr><td></td><td></td><td colspan="2">

- Robert C. Seacord, Effective C: An Introduction to Professional C Programming, 2020, ISBN-13: 978-1-71850-104-1
- Michael Kerrisk, The Linux Programming Interface: A Linux and UNIX System Programming Handbook, No Starch Press; 1st edition (October 28, 2010), ISBN-13: 978-1593272203.

http://index-of.es/OS/The%20Linux%20Programming%20Interface.pdf
</td></tr>
<tr><td>13.</td><td>COURSE SCHEDULE</td><td colspan="2"></td></tr>
<tr><td></td><td>Week</td><td>Instruction Method</td><td>Topic/Activities</td></tr>
<tr><td rowspan="2">1</td><td>Lecture 1, Practical Work 1 (2 hrs). Chp 1</td><td>Introduction; The course syllabus and evaluation system; Class overview and brief summary of labs;</td></tr>
<tr><td>Lecture 2, Practical Work 2 (3 hrs).</td><td>Virtual Machines; Setting up environment for the labs; Introduction to Linux CLI; Practice problems on lecture material</td></tr>
<tr><td rowspan="2">2</td><td>Lecture 3, Practical Work 3(2 hrs). Chp 2.1</td><td>Bits, Bytes and Integers(Part 1); Practice problems on lecture material</td></tr>
<tr><td>Lecture 4, Practical Work 4; Recitation 1; (3 hrs). Chp 2.2-2.3</td><td>Bits, Bytes and Its(Part 2); Practice problems on lecture material; Data Lab Out; Recitation on Data Lab</td></tr>
<tr><td rowspan="2">3</td><td>Lecture 5, Practical Work 5 (2 hrs). Chp 3.1-3.5</td><td>x86-64: Basics; Practice problems on lecture material</td></tr>
<tr><td>Lecture 6, Practical Work 6; Recitation 2; (3 hrs). Chp 3.6</td><td>x86-64: Control; Practice problems on lecture material; Recitation on Bomb Lab; Feedback on Data Lab;</td></tr>
<tr><td rowspan="2">4</td><td>Lecture 7, Practical Work 7(2 hrs). Chp 3.7</td><td>x86-64: Procedures; Practice problems on lecture material</td></tr>
<tr><td>Lecture 8, Practical Work 8; Recitation 3 (3 hrs). Chp 3.8-3.9</td><td>x86-64: Data; Practice problems on lecture materia; Recitation on Debugging, Address Sanitizes and Valgrind</td></tr>
<tr><td rowspan="2">5</td><td>Lecture 9, Practical Work 9 (2 hrs). Chp 3.10</td><td>x86-64: Advanced; Practice problems on lecture material</td></tr>
<tr><td>Lecture 10, Practical Work 10; Recitation 4; (3 hrs). Chp 5</td><td>Code Optimization; Practice problems on lecture material; Recitation on Attack Lab; Feedback on Bomb Lab</td></tr>
<tr><td>6</td><td>Lecture 11, Practical Work 11 (2 hrs). Chp 6.1-6.3</td><td>Memory Hierarchy; Practice problems on lecture material</td></tr>
</table>

		Lecture 12, Practical Work 12; Recitation 5; (3 hrs). Chp 6.4-6.7	***Cache Memories***; *Practice problems on lecture material*; *Recitation on C Programming and Cache Lab*; ***Cache Lab Out***; *Feedback on* ***Attack Lab***
	7	Lecture 13, Practical Work 13 (2 hrs). Chp 7	***Linking***; *Practice problems on lecture material*
		Lecture 14, Practical Work 14(3 hrs). Chp 8.1-8.4	***Exceptions & processes***; *Practice problems on lecture material*; *Exam Review*
	8	**Midterm Exam - 2 hours**	
		Lecture 15, Practical Work 15 (3 hrs). Chp 8.5-8.8	**Signals;** *Practice problems on lecture material*; *Feedback on Exam*
	9	Lecture 16, Practical Work 16(2 hrs). Chp 10	***System Level I/O***; *Practice problems on lecture material*
		Lecture 17, Practical Work 17; Recitation 6; (3 hrs). Chp 9.1-9.6	***Virtual memory: concepts***; *Practice problems on lecture material*; ***Shell Lab out***; *Recitation on Shell Lab*;
	10	Lecture 18, Practical Work 18 (2 hrs). Chp 9.7-9.8	***Virtual memory: systems***; *Practice problems on lecture material*
		Lecture 19, Practical Work 19; Recitation 7; (3 hrs). Chp 9.9	***Storage allocation: basic***; *Practice problems on lecture material*; ***Malloc Lab Outl***; *Recitation on Malloc Lab*; *Feedback on Shell Lab*
	11	Lecture 20, Practical Work 20 (2hrs). Chp 9.9-9.11	***Storage allocation: Advanced***; *Practice problems on lecture material*
		Lecture 21, Practical Work 21; Recitation 8 (3 hrs). Chp 11.1 - 11.4	***Network Programming: Part I***; *Practice problems on lecture material*; *Recitation on Debugging Malloc Lab*
	12	Lecture 22, Practical Work 22 (2 hrs) Chp 11.5-11.6	***Network Programming: Part II***; *Practice problems on lecture material*
		Lecture 23, Practical Work 23; Recitation 9; (3 hrs) Chp 12.1-12.3	***Concurrent Programming***; *Practice problems on lecture material*; ***Proxy Lab Out***; *Recitation on Proxy Lab*;
	13	Lecture 24, Practical Work 24 (2 hrs) Chp 12.4 - 12.5	***Synchronization: Basic***; *Practice problems on lecture material*

		Lecture 25, Practical Work 25; Recitation 10; (3 hrs) Chp 12.5 - 12.7	***Synchronization: Advanced**; Practice problems on lecture material; Recitation On Concurrency in Other Programming Languages*
	14	Lecture 26, Practical Work 26 (2 hrs) Chp 12.6	**Thread-level Parallelism: Part I;** *Practice problems on lecture material*
		Lecture 27, Practical Work 27 (3 hrs) Chp 12.6	**Thread-level Parallelism: Part II;** *Practice problems on lecture material; Recitation on Parallel Programming in Real World; Feedback on* ***Proxy Lab***
	15	Lecture 28, Practical Work 28 (2 hrs)	**Next Steps in Systems programming:** *Operating Systems* **Feedback on Proxy Lab**
		Lecture 29, Practical Work 29 (3 hrs)	**Unix Server Design;** *10K Problem*
	Final Exam – 2 hours		

Information for students:

Plagiarism – It's absolutely unacceptable to use somebody's work, idea or thought in homework, presentation or any other kind of written task, without indicating the source. In this case, the lecturer is obliged to leave the work of a student without assessment.

Cheating – This kind of action is forbidden during any type of activity (homework, exam, presentation, etc.) In this case, the lecturer is obliged to leave the student without assessment.

A Student can retake the mid-term exam in case he/she misses the exam due to a reasonable excuse (illness, business trip, necessity to be at work at a given time). In order to retake the midterm exam, the student must apply to the Faculty of Business, Technology and Education administration. The lecturer fixes the time and date of the retake examination. Involvement in discussions/debate can't be recovered.