

COM 118, 122, 119: Structured Programming, Introduction to Programming, Object-oriented Programming

American University of Central Asia
Software Engineering Department

1 Course Information

Course Codes

COM-118
COM-122
COM-119

Course IDs

4322
5682
4357

Prerequisites for

COM-118:

COM-119, Object-oriented Programming

COM-119:

COM-223, Algorithms and Data Structures
COM-410, Computer Architecture and Organization
COM-421, Software Engineering I
COM-431, Senior Thesis I
COM-433, Senior Thesis II
MAT-407, Numerical Methods
Various elective courses

Credits

6

Professors, Time, Place

Dmitriy Toksaitov

Lecture: Monday 10:50–12:05, 440

Zarlyk Jumabek uulu

Lab: Wednesday 10:50–12:05, Lab G31

Lab: Wednesday 12:45–14:00, Lab G31

Lab: Friday 12:45–14:00, Lab G30

Lab: Friday 14:10–15:25, Lab G30

Pavel Ges

Lab: Tuesday 10:50–12:05, Lab G31

Lab: Tuesday 14:10–15:25, Lab G31

Course Materials

<https://github.com/auca/com.118-119>

2 Contact Information

Professors

Dmitrii Toksaitov

toksaitov_d@auca.kg

Zarlyk Jumabek uulu

jumabek_z@auca.kg

Pavel Ges

ges_pa@auca.kg

TAs

Kamila Inchina

inchina_k@auca.kg

Office

AUCA, room 315

Office Hours

By appointment throughout the work week (write to your professor or TA to make an appointment)

3 Course Overview

This course helps to equip students with essential skills needed for structured and object-oriented programming. At the completion of the course, students should understand fundamental programming concepts such as memory management, flow control, methods, procedural decomposition, objects, classes, inheritance, and polymorphism; be able to write simple applications using most of the capabilities of the Java programming language and apply principles of good programming practices throughout the process.

At the end of the course, the student should be able to research, analyze, design, develop, and maintain functioning software systems according to the goals of the AUCA Software Engineering Department and the 510300 IT competency standard.

4 Topics: Structured Programming

- Week 1–2: Introduction to the Process of Software Development (6 hours)
- Week 3–5: Selections (9 hours)
- Week 6–9: Loops (12 hours)
- Week 10–12: Methods (9 hours)
- Week 13–14: Single- and Multidimensional Arrays (6 hours)

5 Topics: Object-oriented Programming

- Week 1–3: Objects and Classes (9 hours)
- Week 4–6: Inheritance and Polymorphism (9 hours)
- Week 7–8: Abstract Classes and Interfaces (6 hours)
- Week 9–10: Exception Handling (6 hours)
- Week 11–12: GUI and Computer Graphics Basics (6 hours)
- Week 13–14: Generics and Container Classes (6 hours)
- Week 15–16: Working with I/O (6 hours)

6 Assignments and Exams

6.1 GitHub Checkpoints

Students will have to maintain a personal private GitHub repository with all their works shared with their instructor. Students have to periodically commit and push a specific number of lab and URI solutions (or project parts) as told by the instructor. The instructor and TAs will regularly check the work and give points for the accomplished work.

6.2 Labs, URI Tasks, Projects

Students will have eight laboratory tasks, get many problems from an Online Judge System, and finish two projects. Finally, students will have to defend their work during midterm, final, and project defense examination sessions.

7 Course Materials, Recordings and Screencasts

Students will find all the course materials on GitHub. We hope that by working with GitHub, students will become familiar with the Git version control system and the popular (among developers) GitHub service. Though version control is not the focus of the course, some course tasks may have to be submitted through it on the GitHub Classroom service.

Every class is screencasted online and recorded to YouTube for students convenience. An ability to watch a class remotely at any time **MUST NOT** be a reason not to attend classes. Active participation is necessary to succeed in this course.

8 Software

Students are recommended to install the following software on their machines.

- Eclipse Temurin JDK 17: <https://adoptium.net>
- IntelliJ IDEA Community Edition: <https://www.jetbrains.com/idea/download>
- Git: <https://git-scm.com>

9 Reading

Intro to Java Programming, Comprehensive Version, 12'th Edition by Y. Daniel Liang (AUCA Library Call Number: QA76.73.J38 L52 2020, ISBN-13: 978-0136520238, ISBN-10: 0136520235)

10 Grading

10.1 GitHub Checkpoints

Your instructor will announce a periodic review of your work. You will be awarded up to the following number of points for such checks.

- Labs (10%)
- Online Judge Problems (10%)
- Project #1 (4%)
- Project #2 (4%)

10.2 Exams and Project Defense

- Labs 1–4 (4% + 4% + 4% + 4%)
- Online Judge Problems (10%)

Midterm Exam (Labs + Online Judge Problems)

- Labs 5–8 (4% + 4% + 4% + 4%)
- Online Judge Problems (10%)

Final Exam (Labs + Online Judge Problems)

- Project #1 (10%)
- Project #2 (10%)

Project Defense (Both projects will be checked during or after the Final Exam)

10.3 Totals

- 100% is formed from the GitHub submissions (28%) and the exam sessions (72%).

10.4 Scale

- [92%–100] %: A
- [85%–92) %: A-
- [80%–85) %: B+
- [75%–80) %: B
- [70%–75) %: B-
- [65%–70) %: C+
- [60%–65) %: C
- [55%–60) %: C-
- [50%–55) %: D+
- [45%–50) %: D
- [40%–45) %: D-
- Less than 40%: F

Please, note that requests to award a better grade if the number of points is close to such a grade will be ignored. For example, 91.99 is A-, NOT A. Likewise, requests

to get extra assignments to increase the number of points will also be overlooked entirely.

11 Rules

Students are required to follow the rules of conduct of the Software Engineering Department and the American University of Central Asia.

11.1 Participation

Active work during the class may be awarded with extra points at the instructors discretion. Poor student performance during a class can lead to points being deducted from the final grade.

Instructors may conduct pop-checks during classes at random without prior notice. Students **MUST** be ready for every class in order not to lose points. Students absent without a good reason from such classes with graded work will also lose points unless it is force-majeure circumstances. TAs and instructors must be notified in advance about why a student is absent not to lose points.

11.2 Questions

We believe that a question from one student is most likely a question that other students are also interested in. That is why we encourage students to use the online discussion board of the LMS (Learning Management System) that you use (e.g., AUCA e-Course System) to ask questions in public that other students can see and answer. We discourage students from asking questions through E-mail. If it is a private matter, write direct messages to TAs or your instructor through the LMS system (such as AUCA e-Course System) too. We will not be answering most E-mail messages this semester (unless it is a severe emergency) to consolidate all the course correspondence in one place.

Do not post the complete source code for any task on the LMS discussion board. You will get zero for that work for any such public post. Do not ask generic questions about your code to know why it does not work. Please spend some time thinking about your code, debugging it.

11.3 Late Policy

Late submissions and late exams are not allowed. Exceptions may be made at the professor's discretion only in force-majeure circumstances. If you got ill, got severe personal issues, got problems with your computer or the Internet, you **MUST** notify instructors at least 24 hours in advance. Otherwise, we will not give you an extension. We will consider that you were procrastinating until the very last day. We will also not be giving more than one emergency extension throughout the course.

Six hours before the deadline for any work on the course, instructors will go into a silent mode. No questions will be answered about the work that has to be

submitted, no requests to have office hours will be considered. Usually, it will be Saturday and Sunday (which are not working days for us anyway). However, at any other work time before the deadline, we will try our best to answer your questions and help you through Zoom or in our office.

11.4 Exam Ceremonies

Students **MUST** follow exam ceremonies. It means they **MUST** prepare task list forms with all points appropriately calculated. They **MUST** submit them correctly. They must bring task list forms to the exam. Failure to do so will result in lost points. Throughout your career, you will have to work with various supporting documents (contracts, timesheets, etc.). It is a good idea to start learning to work with such documents accurately early. We will remove points for not following these rules or even refuse to accept your exam defense. We will also give zero for not following the strict exam timing rules.

11.5 Incomplete Grade

As with late exams, the grade *I* may be awarded only in exceptional circumstances. The student must start a discussion on getting the grade *I* with the instructors in advance and not during the last week before the final exams.

11.6 Academic Honesty

Plagiarism can be defined as an act or an example of copying or stealing someone elses words or ideas and appropriating them as ones own. The concept of plagiarism applies to all tasks and their components, including program code, comments, documentation, abstracts, reports, graphs, statistical tables, etc.

The following are examples of some common acts of plagiarism:

1. Representing the work of others as their own
2. Using other people's ideas or phrases without specifying the author
3. Copying code snippets, sentences, phrases, paragraphs or ideas from other people's works, published or unpublished, without referring to the author
4. Replacing selected words from a passage and using them as your own
5. Copying from any type of multimedia (graphics, audio, video, Internet streams), computer programs, graphs or diagrams from other people's works without representation of authorship
6. Buying work from a website or from another source and presenting it as your own work

In addition to being unethical, this indicates that the student has not studied the given material. Tasks written from somewhere for 5% or less will be assessed

accordingly or will receive a 0 at the discretion of the teacher. If plagiarism is more than 5%, the case will be transferred to the AUCA Disciplinary Committee.

In this course, teamwork is NOT encouraged. The same blocks of code or similar structural pieces in separate submissions will be considered academic dishonesty, and all parties will get zero for the task.

Students are not recommended to memorize lab and project code before exams, as this is a difficult and inefficient way to learn; and since practice exams may consist of open questions designed to test a students analytical skills, memorization invariably leads to the fact that the answers are inappropriate and of poor quality.