COM 118, 119: Structured Programming, Object-oriented Programming

American University of Central Asia
Software Engineering Department

1 Course Information

Course Codes

COM-118
COM-119

Course IDs

4322

Prerequisite for

COM-119, Object-oriented Programming
or
COM-223, Algorithms and Data Structures
COM-410, Computer Architecture and Organization

Credits

6

Professors, TAs, Time, Place

Lecture (Dmitrii Shostak): Wednesday 12:45–14:00, Online
Lab (Dmitrii Shostak): Wednesday 14:10–15:25, Online
Lab (Dmitrii Shostak): Thursday 10:50–12:05, Online
Lab (Dmitrii Shostak): Thursday 12:45–14:00, Online
Lab (Dmitrii Shostak): Thursday 14:10–15:25, Online
Lecture (Dmitrii Toksaitov): Tuesday 10:50–12:05, Online
Lab (Dmitrii Toksaitov): Tuesday 12:45–14:00, Online
Lab (Dmitrii Toksaitov): Tuesday 14:10–15:25, Online
Lab (Dmitrii Toksaitov): Thursday 12:45–15:25, Online
Lab (Dmitrii Toksaitov): Tuesday 14:10–15:25, Online

Course Repository

`https://github.com/auca/com.118-119`

Class Discussions

`https://piazza.com/auca.kg/fall2020/com118/home`
`https://piazza.com/auca.kg/fall2020/com119/home`

Student Informal Discussions

`https://auca-sfw.slack.com/archives/G019KJBUTB7`

2 Contact Information

Professors

Dmitrii Shostak
shostak_d@auca.kg
Dmitrii Toksaitov
toksaitov_d@auca.kg

Office

AUCA, room 315

Office Hours

By appointment throughout the work week remotely through Zoom

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 flow control, objects, classes, methods, procedural decomposition, 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 student should be able to research, analyze, design, develop, and maintain functioning software systems in accord to the goals of the AUCA Software Engineering Department and the 510300 IT competency standard (OK 17, 17, 115).

4 Topics Covered

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)

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)

5 Assignments and Exams

5.1 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 will periodically check the work and give points for the accomplished work.

5.2 Labs, URI Tasks, Projects

Students will have eight laboratory tasks, get a number of problems from an Online Judge System, and have to finish two projects developing real-world applications. Students will have to defend their work to the instructor during midterm, final, and project defence examination sessions.

6 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 the online Zoom session. Active class participation is necessary to succeed in this course.

7 Software

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

- Amazon JDK 8: `https://aws.amazon.com/corretto`
- IntelliJ IDEA: `https://www.jetbrains.com/idea/download`
- Git: `https://git-scm.com`

8 Reading

Intro to Java Programming, Comprehensive Version, 10th Edition by Y. Daniel Liang (AUCA Library Call Number: QA76.73.J38 L5218 2011, ISBN-13: 978-0133761313, ISBN-10: 0133761312)

9 Grading

9.1 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%)

9.2 Interview

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

Midterm Defense (Labs + Online Judge Problems)

- Project #1 (10%)

Project Defense (As a separate exam after the Midterm)

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

Final Defense (Labs + Online Judge Problems)

- Project #2 (10%)

  Project Defense (As a separate exam after the Final)

9.3 Total

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

9.4 Scale

- 92%–100%: A
- 85%–91%: A-
- 80%–84%: B+
- 75%–79%: B
- 70%–74%: B-
- 65%–69%: C+
- 60%–64%: C
- 55%–59%: C-
- 50%–54%: D+
- 45%–49%: D
- 40%–44%: D-
- Less than 40%: F

10 Rules

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

10.1 Participation

Active work during the class may be awarded with up to 5 extra points at the instructors discretion.

Poor student performance during a class can lead to up to 5 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.

10.2 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 Piazza to ask questions in public that other students can see and answer and NOT ask them through E-mail in private UNLESS the question itself is about private matters to discuss with the professor.

10.3 Late Policy

Late submissions and late exams are not allowed. Exceptions may be made at the discretion of the professor only in force-majeure circumstances.

10.4 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.

10.5 Incomplete

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.

10.6 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, abstracts, reports, graphs, statistical tables, etc.

In addition to being unethical, this indicates that the student has not studied the given material. Tasks written from somewhere for 10% or less will be assessed accordingly or will receive a 0 at the discretion of the teacher. If plagiarism is more than 10%, the case will be transferred to the AUCA Disciplinary Committee.

Students are not recommended to memorize before exams, as this is a difficult and inefficient way to learn; and since practice exams 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.

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

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