COM-119: Object-oriented Programming

American University of Central Asia
Software Engineering Department

1 Course Information

Course Codes
COM-119

Course IDs
4357

Prerequisites for

COM-223, Algorithms and Data Structures
COM-410, Computer Architecture and Organization

Credits
6

Professors, Time, Place

Dmitrii Toksaitov
Lecture: Tuesday 11:50–13:05, Room 432
Lecture: Thursday 11:50–13:05, Room 432
Lab: Tuesday 13:05–14:20, Room 432
Lab: Thursday 13:05–14:20, Room 432

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

2 Contact Information

Professors
Dmitrii Toksaitov
toksaitov_d@auca.kg

Office
AUCA, room 315

Office Hours
By appointment throughout the work week (write to your professor 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 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.

4 Topics

- Week 1–2: Elementary Programming (10 hours)
- Week 3: Selections (5 hours)
- Week 4: Loops (5 hours)
- Week 5: Methods (5 hours)
- Week 6: Single- and Multidimensional Arrays (5 hours)
- Week 7: Objects and Classes (5 hours)
- Week 8: Inheritance and Polymorphism, String Processing (5 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 URI solutions as told by the instructor. The instructor will regularly check the submissions and give points for the accomplished work.

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. The course Zoom link is `http://bit.do/auca-com-119`. YouTube links

can be found on the course repository at `https://github.com/auca/com.118-119`. An ability to watch a class remotely at any time MUST NOT be a reason not to attend at least 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.

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

8 Reading

Introduction to Java Programming, Comprehensive Version, 12'th Edition by Y. Daniel Liang (ISBN-13: 978-0136520238, ISBN-10: 0136520235)

9 Grading

9.1 GitHub Checkpoints

Your instructor will announce a periodic review of your work. You will be awarded a certain number of points for such checks.

- 100% is formed from the GitHub submissions (10%) and in-class work defence (90%).

9.2 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.

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 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 Canvas online discussion board 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 your instructor on Canvas. 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 on our LMS (Learning Management System).

Do not post the complete source code for any task on the Canvas 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.

10.2 Late Policy

Late submissions 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 me at least 12 hours in advance. Otherwise, we will not give you an extension.

10.3 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 of the course.

10.4 Academic Honesty

Plagiarism can be defined as "an act or an example of copying or stealing someone else's words or ideas and appropriating them as one's 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 code before defence, as this is a difficult and inefficient way to learn; and since practice defence may consist of open questions designed to test a student's analytical skills, memorization invariably leads to the fact that the answers are inappropriate and of poor quality.