COM 121, Safety in Programming Languages

American University of Central Asia
Software Engineering Department

1 Course Information

Course Codes
COM-121

Course IDs
5291

Prerequisites for
COM-223, Algorithms and Data Structures

Credits
3

Professors, Time, Place

Dmitrii Toksaitov
Lab: Monday 15:35–16:50, Lab 432
Lab: Tuesday 14:10–15:25, Lab 432

Azamat Derkenbaev
Lab: Wednesday 10:50–12:05, Lab 432
Lab: Wednesday 12:45–14:00, Lab 433
Lab: Wednesday 14:10–15:25, Lab 433

Course Materials
`https://github.com/auca/com.121`

2 Contact Information

Professors
Dmitrii Toksaitov
toksaitov_d@auca.kg
Azamat Derkenbaev
derkenbaev_a@auca.kg

TAs
Artur Talkanbaev
talkanbaev_a@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

Safety in Programming Languages is an introductory course covering the most important aspects of Software System Safety from a developer's point of view. The 3-credit course focuses on topics of building secure, safe, and sustainable software. This course takes an in-depth look into low-level programming languages such as C, C++, and assembly, their relationship with the underlying hardware. We try to understand how design decisions of such foundational technologies, used to build core systems around us may influence virtual and real-world sustainability and safety.

At the end of the course student should be able to research, analyze, design, develop, and maintain functioning software systems in C in accord to the goals of the AUCA Software Engineering Department and the 510300 IT competency standard (OK 1–7, 1–7, 1–15).

4 Topics

- Week 1: Tools and Development Environment (3 hours)
- Week 2–3: The Compilation Process of C Programs, Build Systems (6 hours)
- Week 4–5: Safe vs. Unsafe, Portability (6 hours)
- Week 6–11: Structured and Procedural Programming in C: Conditionals, Loops, Functions (15 hours)
- Week 12–13: Memory Management in C, Pointers, Arrays (6 hours)
- Week 14–15: The World of C++, Basics of OOP in C++ (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 solutions as told by the instructor. The instructor and TAs will regularly check the work and give points for the accomplished work.

5.2 Students' Recordings

Students will have to maintain a private Google Drive directory to keep recordings of the experiments they do. Links to recordings will have to be submitted together with lab solutions on GitHub. These recordings will also be graded.

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-121`. YouTube links can be found on the course repository at `https://github.com/auca/com.121`. 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.

- Git: `https://git-scm.com`
- Microsoft Visual Studio Code: `https://code.visualstudio.com`

The compilers, assemblers, and debuggers will be available on the remote course server.

8 Reading

- C Programming: A Modern Approach, Second Edition, by K. N. King (AUCA Library Call Number: QA76.73.C15 K49 2008, ISBN: 978-0393979503)
- The C Programming Language, Second Edition, by Brian Kernighan and Dennis Ritchie (AUCA Library Call Number: QA76.73. C15 K47 1988, ISBN: 978-0131103627)

9 Grading

This 3-credit-hour course will not have any exams. Instead, your instructor will announce a periodic review of your work. You will be awarded a certain number of points for such checks. The points will form your final course grade.

9.1 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 Participation

Active work during the class may be awarded with up to 5 extra points at the instructor's 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. 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.

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 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 TAs or your instructor there on Canvas 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 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.

10.3 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 the 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.

10.4 Incomplete

As with late submissions, 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.

10.5 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 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 student's analytical skills, memorization invariably leads to the fact that the answers are inappropriate and of poor quality.