BPI1116 Programming technologies
Course description

Preliminaries: well-founded knowledge of object oriented programmming in Java
Final target: practical grade
Lecturer:Vályi Sándor Zoltán (PhD) assoc. prof., valyi.sandor[@]nye[dot]hu
Time period of the course: Tuesdays 10:00 – 13:10
Consultation time period: Tuesdays 13:30 – 14:20, room B220
Mid-year requirements, conditions of the practical grade:

1. solving home work problems and upload your solution for weeks 1–8,10,11 [10*5p, minimum 2p for each week]
2. paper test for Java language and Java OO questions [20p, minimum 10p]
3. paper test for Java design pattern questions [20p, minimum 5p]
4. projekt work and oral defense [60p, minimum 20p].

Rating: each minimum point is provided AND 55 p – sufficient, 80 p – medium, 100 p – good, 130 p -- excellent
Participation on classes:
The minimum home work points make valid the participation. These points will be registered on the classroom.google.com course.
Schedule of assessments: home work – Monday 23:59 after the class. Projekt work submit: 24 hours before the defense.

Organization: The classes will be held in person in rooms E14/15, but will also be broadcast on MS Teams in the PRtEN25 group, of which the students are already members. Moreover, recordings will be made, which can be viewed until the end of the course. It will be advantageous to use your own laptop, most of the preparation will be done at home, in the form of homework. Registration for participation is done by upload the solution of the homework assignments submitted for the course on classroom.google.com for the given week, within the given time limit. However, to obtain at least a sufficient grade, at least 2 points are always required, in each given week. (week 1--8,10,11).

Home page for this course: https://classroom.google.com/c/Nzc1MTc2MDIwNDQ5?cjc=hdpsz2pb

Topics/Schedule based on weeks (week 1: 8th September 2025..week 14: 9th December same year):

1. Tools, installation, build tools. Exercise: construct an own maven project *[5p]*
2. version control, git, github. Exercises: do some commits, do a branching *[5p]*
3. Software crysis and solutions. Structured and modular programming. Using an IDE. Object oriented programming in Java – repetition. Exercise: a Java OO problem set, part I. *[5p]*
4. Frequently used Java classes. Generic programming. Java Collections and data structures. Exercise: a Java OO problem set part II. *[5p]*
5. Lambda expressions, Functional programming in Stream API. Exercises: using streams *[5p]*
6. Unit és module testing, JUnit. Exercises: testing your solutions to problems in lessons 2—4. *[5p]*
7. Debugging. Exercise: what is wrong in this code?*[5p]* **Paper test on Java programming [20p] 21st October**
8. Logging. java.util.logging. Using JavaDoc. Exercise: JavaDoc and logging in the solution for lessons 1--3 [*5p*]
9. Example for project work

10. Design Patterns. Exercises: implementation of d. p.-s [5p]
11. Java Database Connectivity. Exercises: using JDBC [5p]
12. Consultation on project work. **Paper test on design patterns. [20p] 25th November**
13. **Project work defense [60p] 2nd December**
14. Spare

Project work:
a Java program with a command interface, which is a 2-person game board editor and game play for a human user against a machine opponent who randomly chooses a move among the possible moves. It saves the edited board with a local SQL database connection. Include exhaustive testing for 'business' logic. Logging for the games. Include JavaDoc developer documentation, at least for each method.
This year's actual 2-person game is go-moku on a NxM rectangular board.
https://en.wikipedia.org/wiki/Gomoku
Project evaluation:
defense: if the student does not know her/his own program, the point will be 0 independently the quality of the code.
using git and maven: 5p,
quality of OO implementation (using objects, methods) 20 p,
correct game play 10p,
tested correctness 15p,
saving into relational db 5p,
showing 2-3 design patterns present in the project source 5p.
alltogether 60 points.