Gebze Technical University
Computer Engineering Department

CSE443 - Object Oriented Analysis and Design
Fall 2019-2020

Homework 1 – v1

Rule 1: no plagiarism (from colleagues or other sources). Detected cases of plagiarism will lead to a significant penalty of your course grade at the end of the semester.

Rule 2: no late submissions! Even if it is late by one minute, it will be ignored. Learning to plan your schedule according to deadlines is part of your education and an invaluable professional asset.

What to submit: a) the source code of your project *fully documented (with javadoc)*, b) a nicely formatted pdf report of your design decision explanations and class diagrams and c) an executable demo that fully illustrates your program's capabilities *whenever code is requested*.

****
Question 1 (25 points): You have been hired by the company "Numerical Solutions Inc" and you are in charge of their new project "LinearSolverDeluxe".

The project's objective is to design and implement a Java application with a single functionality: it admits as input from the user a system of linear equations and outputs its solution, if it exists, or an error message otherwise.

The customer:
- wants the software to support at least two methods of solving linear equations: Gaussian elimination and matrix inversion.
- wants to be able to change between solving methods dynamically.
- might need more functionalities in the future (e.g. determinant calculation, etc).

You will:
- Provide the class diagrams with the appropriate arrows and contents (in detail; all the methods and variables involved with the appropriate access rights), that satisfies the customer's requirements; strive for maximum flexibility, loose coupling and minimize maintenance costs!
- Implement in Java the above software (user interfacing technique, way of admitting input, way of outputting, that's all up to you).

Question 2 (25 points): after the last project, you were fed up with numerical analysis, and on the spur of the moment, you quit your job. After a few weeks of lurking at your sofa and watching depressive netflix series, you get a brilliant idea that no one has ever thought of before!

Everybody has their own list of favorite websites. But nobody wants to check them daily n times to see if they have been updated with new content (*content = either text, photographs or audio*)! Wouldn't it be great if you could subscribe to your favorite website, and the websites somehow notified the subscribers of new content? So much wow, such originality! Doge is impressed. You're certain that you'll make millions from this and all major websites will support it!

However, as a seasoned professional (who has never heard of RSS) you must make sure that:
- it's trivial to add and remove subscribers, as it'll happen often

- a subscriber can be in the form of any software using your library. So your API must be easy to use.
- a subscriber might be interested in only new text updates, or photograph updates or audio updates *or a combination thereof*. There is no need to disturb them if the update is not of the desired type. What kind of a design would support this?
- What if users or websites demand your software to support a fourth type of content? Will it be easy to modify? It better be.

You will:
- Provide the class diagrams with the appropriate arrows and contents (in detail, all the methods and variables involved with the appropriate access rights), that satisfy the design requirements; strive for maximum flexibility, loose coupling and minimum cost of maintenance!

Question 3 (25 points): The last project didn't do well on the market. You quit freelancing and you are now employed by ZırhSan A.Ş, a private company specializing in the design and production of exoskeleton armored suits for military personnel, equipped with various custom weapons.

There are 3 basic types of suits: **dec** (500k TL, 25kg), **ora** (1500k TL, 30kg) and **tor** (5000k TL, 50kg).
Each of these suits can be equipped with the following accessories:

- Flamethrower (50k TL per item, 2k)
- AutoRifle (30k TL per item, 1.5kg)
- RocketLauncher (150k TL per item, 7.5kg)
- Laser (200k TL per item, 5.5kg)

A customer can demand any custom combination of accessories: such as a dec suit with 1 flamethrower, 2 automatic rifles and 1 rocket launcher.

Your task is to develop a piece of software in Java *able to calculate the total cost and weight of an equipped suit* (total price = basic suit price + prices of the accessories). The user of the software should be able to designate any combination of accessories *dynamically* at runtime. Your design should be flexible and easy to accommodate new accessories and suit types.

Question 4 (25 points): You quit your job at ZırhSan, as the schedules were hectic (initially advertised as a "esnek çalışma saatlerine sahip pozisyon, bıdı bıdı") and you are now employed by Turkish Aerospace Industries (TAI). TAI is about to launch the new series TPX series of passenger planes, consisting of 3 models.

Model	**Purpose**	**Skeleton**	**Engine**	**Seating**
TPX 100	Domestic flights	Aluminum alloy	Single jet engine	50 seats

TPX 200	Domestic and short international flights	Nickel alloy	Twin jet engines	100 seats
TPX 300	Transatlantic flights	Titanium alloy	Quadro jet engines	250 seats

The production of a plane is conducted in the following order:
1) `constructSkeleton()`
2) `placeEngines()`
and 3) `placeSeats()`

Use the Factory Method design pattern to develop a Java program simulating the production of TPX planes. **(10 points)**

Business is booming. TAI is now exporting planes to the international market. And the models are customized according to local needs.

Market	**Engine injection type**	**Seating cover**	
Domestic	Turbojet	Velvet	
Eurasia	Turbofan	Linen	
Other	Geared turbofan	Leather	

Redesign and re-implement your Java program simulating the production of TPX planes for all 3 markets, using this time the Abstract Factory design pattern (15 points).