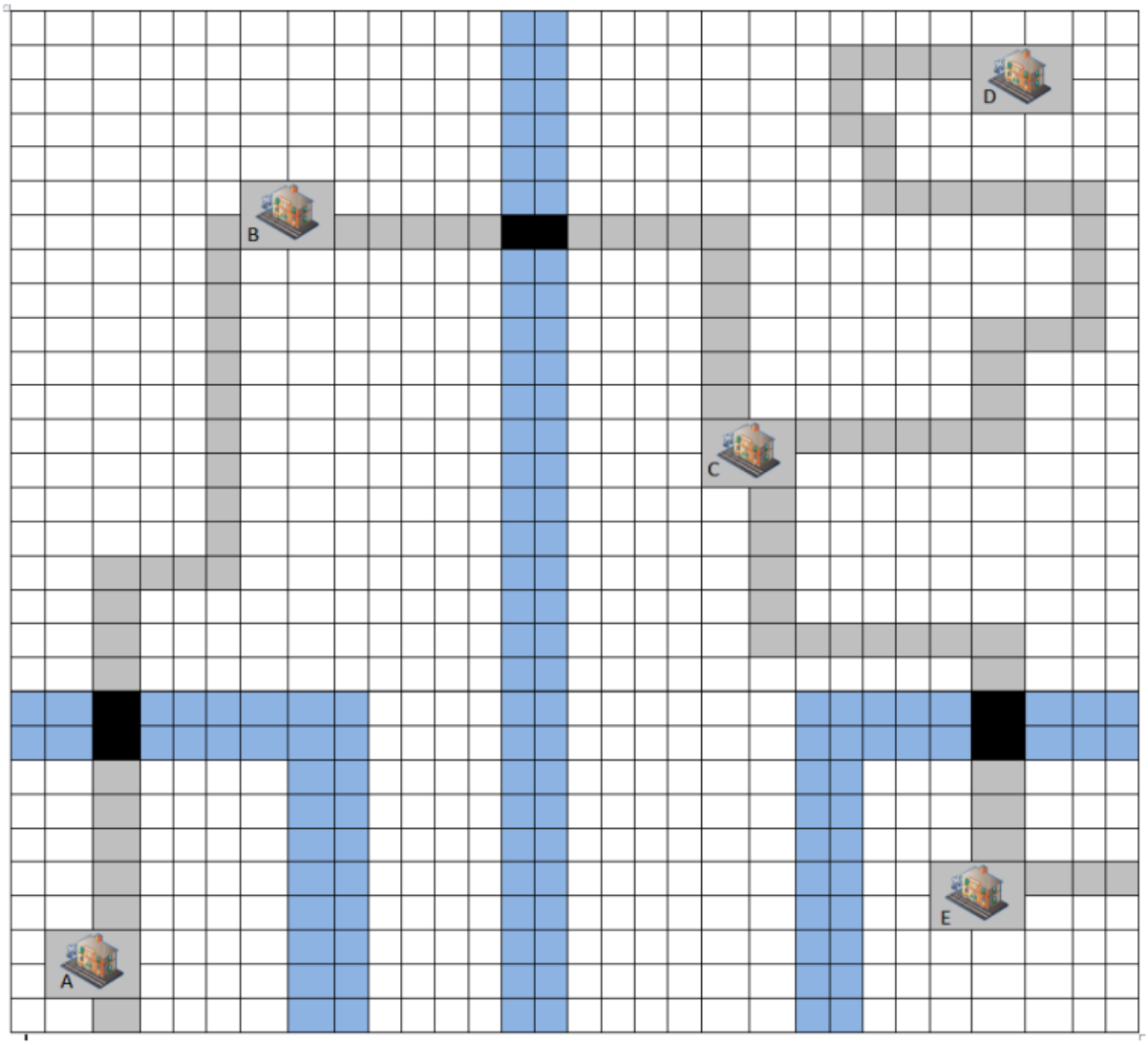
Create a simple simulation of railway traffic. The image shows a map of the territory where railway traffic is conducted. The map is represented by a matrix/table with dimensions 30x30.



On the map, there are railway stations (A, B, C, D, E), railway tracks marked in gray, and roads marked in blue. Black dots represent level crossings. It is implied that there are multiple tracks at the railway stations, while there is only one track on other railway lines.

Railway compositions, consisting of one or more locomotives and zero or more wagons, travel on the tracks. Locomotives can be passenger ones, transporting passenger compositions, freight ones, carrying freight compositions, universal ones, capable of carrying both passenger and freight compositions, and shunting locomotives used for maintaining the tracks. According to the type of propulsion, locomotives can be steam, diesel, or electric. When forming a composition, ensure that locomotives of the same type are used (passenger-passenger, passenger-universal, freight-freight, freight-universal). Each locomotive has a power rating and a designation.

Wagons can be passenger, freight, or special-purpose wagons. Passenger wagons can have seats, beds, sleeping compartments, or dining cars. For wagons with seats and beds, the

number of seats/beds should be indicated, and for dining cars, a description of the facility is needed. Freight wagons have a maximum load capacity. Each wagon has a length and a designation.

Compositions move exclusively on the tracks, and they must not know the paths they need to take. They are only allowed to be aware of the name of their destination station. Compositions are responsible for detecting the tracks within their territory and navigating along them. The tracks are divided into segments between railway stations, and traffic can occur in both directions on all tracks but not simultaneously. The role of railway stations is to monitor the presence of trains in the segments they cover, and compositions themselves must not directly communicate with other trains or be aware of their movements. For all information about the segments of the track they are traveling on, trains communicate with the railway station they are heading towards.

Each train has its own speed, representing the time it spends on one field before moving to the next field on the map. The minimum speed is 0.5. For compositions with electric locomotives, the fields must be electrified, including the field in front of the composition, all fields occupied by the composition, and the field behind the composition.

Railway stations can be used to hold trains if there is a train on the next segment traveling in the opposite direction. For example, if train x is traveling from station B to station C, and another train y is already on the segment heading towards station B, train x waits at station B until train y arrives. If no other train has started behind y towards station B in the meantime, train x can continue its journey. On a segment, there can be multiple trains traveling in the same direction, but their speeds need to be synchronized to avoid collisions (only on that segment, they can return to their initial speed). Train movements on the map are represented by appropriate symbols or images, and it is necessary to emphasize the length of the composition (number of locomotives and wagons). Railway station C is also a point where trains can switch to another track, so, in addition to the described tasks, this station directs trains in the appropriate direction.

Vehicles are used to travel on roads and can be either cars or trucks. Each vehicle has a brand, model, and year of manufacture. Cars have a number of doors, while trucks have a load capacity. Vehicles move at a specific speed, which is lower than the maximum speed defined for each road in the configuration file.

When approaching a level crossing, vehicles stop at a free position if there is a train moving towards them or passing through the crossing at that segment. Vehicles resume their movement only when the train has passed and the level crossing is clear, and the fields are no longer electrified. Only one vehicle can occupy a single field at a time, so it is essential to manage field occupancy to prevent collisions. If the number of vehicles exceeds the available fields, the excess vehicles can be stored off the map. Vehicles move bidirectionally on the roads, and the width of the road is 2 fields.

It is necessary to ensure that vehicles and railway compositions move simultaneously without direct programmatic control or a predictable algorithm.

A configuration file is used to store the following parameters: maximum allowed speed on individual roads, the number of vehicles on each road, and paths for storing files (folders).

Upon launching the program, a map is displayed, and there is an option to start the simulation. When the simulation begins, vehicles are created with predefined speeds generated randomly, respecting the speed limits of the roads they are meant to travel on. They are placed at the starting point of the road they should follow. The direction in which the vehicle moves is randomly determined, and based on this, it is placed at the beginning or end of the road. Vehicles are created individually with a time interval of 2 seconds, and the number of vehicles created on each road is defined in the configuration file. If the configuration file is manually changed during the simulation (through a text editor), the new speed values are used when generating subsequent vehicles. If the number of vehicles is increased, new vehicles are added to the simulation in the same way as the previous ones. If a smaller value is set, the change is ignored.

During program execution, the "vehicles" folder is monitored, where the lines along which the trains move are defined. A way to define the appearance of the composition (number and type of locomotives and wagons, and their arrangement), speed, starting, and destination points needs to be devised in textual form. Each line is defined in a separate .txt file using any text editor. When the program detects a new line, it verifies the configuration's validity and creates a new composition based on that. It sets the composition at the starting point and initiates movement. While moving, each composition creates its movement history, consisting of a series of points it crosses, information about stops at stations, and travel time. These records are serialized in the "movements" folder when the composition reaches the destination station. In the GUI part of the application, an option should be added that opens a new window displaying all movements and their details. It is essential to use the Logger class for handling exceptions in all classes.