OBJECT-ORIENTED PROGRAMMING
Winter semester of the academic year 2021-2022 Course Assignment

Simulate a hypothetical operation (one direction) of the Attiki Odos (Attica Highway) using C++. According to this simulation, the following rules apply to the highway:
Attiki Odos has an initial node, a final node, and a set of intermediate nodes. Only vehicle entry is allowed at the initial node, and only vehicle exit is allowed at the final node. Both entry and exit of vehicles occur at each intermediate node.
Attiki Odos consists of NSegs segments between its nodes. The nodes are in the count of NSegs + 1, from node 0 (initial) to node NSegs (final). Each segment of the highway has a capacity, possibly different for each segment, which represents the maximum number of vehicles that can be inside the segment.
Except for the final node, each node has a number of toll booths for entry to the segment starting from that node. The number of toll booths can differ for different nodes. At each node, some of the toll booths are operated by collectors, while others are electronic.
In a state of the highway, there is a number of vehicles present in each segment. Some of these vehicles are in a "ready" state to exit the segment. Among them, those destined for the exit node of the segment will exit the highway in the next state. Those destined for a subsequent exit node must eventually move to the next segment, which may become feasible in the next state. The remaining vehicles not in a "ready" state will remain in the same segment in the next state. There is also a number of vehicles waiting to enter the segment from the entry toll booths. Each toll booth (either operated by a collector or electronic) can have a different number of vehicles waiting to enter it. From the vehicles present in the segment in the new state (those present in it during the previous state, those that passed from the previous segment if applicable, and those that entered from the entry node of the segment), a percentage Percent of these vehicles transitions to a "ready" exit state from the segment.
To create the next state of the highway, the simulation works from the end towards the beginning. Vehicles destined for the final node of the highway will exit it. Then, the vehicles that are in a "ready" state from the immediately previous segment to pass into the current one will do so, as long as it is allowed by the segment's capacity. If it's not possible for all of them to pass, only enough will pass to not violate the capacity. If, after the vehicles from the previous segment have entered the current segment, there is availability for new vehicles to enter, this will happen from the entry toll booths. A maximum of K vehicles can enter from toll booths with collectors and a maximum of 2K vehicles can enter from electronic toll booths, while still respecting the segment's capacity. This means that a percentage, the same for all toll booths, of vehicles from the K and 2K allowed to enter will be allowed to enter the segment in its current state (the current simulation cycle). If fewer vehicles need to enter than the maximum allowed, K will be decreased by 1 for the specific node in the next simulation cycle. If the maximum number of vehicles, K and 2K for the two types of toll booths, enter the segment, K will be increased by 1 for the specific node in the next cycle.
Once it's determined which vehicles will be present in a segment of the highway in the new state, based on what will happen in the next segment, the simulation proceeds to apply the above procedure to the previous segments gradually until reaching the initial node.

In a section scenario:

• If the number of vehicles entering a section from its entrance is smaller than the number of vehicles waiting at the entrance, the message "Delays at the entrance of node 'Node Number'" is displayed.
• If (additionally) the number of vehicles moving between the connected sections of the node is smaller than those in a state of "ready" to move, the message "Delays after node 'Node Number'" is also displayed.
• If none of the above conditions apply, the message "Maintain safety distances in the section after node 'Node Number'" is displayed.
Specifically, the Attiki Odos (Highway) is represented by a class that:
• Has NSegs sections.
• Has a total number of vehicles moving within it.
Its behavior is as follows:
• Initially, the message "Highway in operation" is printed.
• The operation of the highway consists of cycles of operation for its sections. Sections operate sequentially, starting from the last one. Each time, the number of vehicles is displayed. (operate)
A segment of the highway is represented by a class that:
• Has an entrance.
• Has a set of vehicles moving within it.
• Has a vehicle capacity.
• Has (possibly) a next section.
• Has (possibly) a previous section.
Its behavior is as follows:
• Initially, random vehicles moving within it are created.
• Vehicles that are possible to enter the section do so. (enter)
• Vehicles that can exit the highway do so. (exit)
• Vehicles that can exit the section and move to the next one do so. (pass)
• The number of vehicles moving within it is returned. (get number of vehicles)
• It operates, changing the set of vehicles moving within it. In case it's not possible for all the vehicles ready to exit to move to the next section, the selection is done randomly. Additionally, when changing the set of vehicles, incoming vehicles are removed from its entrance. During its operation, relevant messages are displayed. Also, some (random) vehicles are marked as ready to exit the section. (operate)
An entrance is represented by a class that:
• Has the name/identity of the node.
• Has a set of toll booths with staff.
• Has a set of electronic toll booths.
Its behavior is as follows:
Vehicles are removed from the entrance, removing vehicles from the toll booths. Random vehicles are added to its toll booths. (operate)
A toll booth is represented by the set of vehicles waiting to enter it. Initially, randomly waiting vehicles are created. Vehicles are added to the toll booth. Vehicles are removed from the toll booth.
Vehicles are represented by their exit node and the section of the highway they are moving within. When waiting to enter the highway, the section indicator is -1. They also include an indicator of readiness to exit the section they are moving within (have traversed). Initially, they are not ready.

Implement the necessary classes in C++. Also, implement a main function that executes N cycles of the highway's operation, during each of which random vehicles are added to the entrances of the highway's nodes.
From the command line, during program execution, the following are provided:
N: Number of simulation cycles.
NSegs: Number of sections in Attiki Odos.
K: Maximum number of vehicles that can enter a highway section from a toll booth with a collector. Initially, K is the same for all toll booths of all nodes (2K for electronic toll booths).
Percent: The percentage of vehicles in a section, common for all sections, that transition to a "ready" state to exit the section to the next state.
The capacities of the sections of Attiki Odos are read from the standard input.
The initial state of Attiki Odos is randomly generated. At each simulation step, the number of vehicles wishing to enter from a toll booth into the highway is randomly selected.
Where needed, you may freely make assumptions about the operation of the highway based on your general experience from the real world and document them.