

## Here Technologies Coding Exercise

### Autonomous Driving Speed Simulator

A driverless vehicle requires a program to determine what the appropriate speed is at which it should drive. You must set the autonomous driving mode when starting your program. When a sensor event number is entered, then output the desired speed. Below is a list of possible driving modes and sensor events with the desired speed or speed modification for each mode.

#### Autonomous Driving Modes

- Normal
- Sport
- Safe

#### Sensor Events

Event	Description	Normal	Sport	Safe
1	Traffic	-10 kph	-5 kph	-15 kph
2	Traffic Clear	+10 kph	+5 kph	+15 kph
3	Weather Rainy	-5 kph	-5 kph	-5 kph
4	Weather Clear	+5 kph	+5 kph	+5 kph
5	Slippery Road	-15 kph	-15 kph	-15 kph
6	Slippery Road Clear	+15 kph	+15 kph	+15 kph
7	Emergency Turbo	+20 kph	+30 kph	+10 kph
10	Speed Limit Sign X	X kph	X+5 kph	X-5 kph

Plus (+) and minus (-) signs indicate an acceleration or deceleration while events without a plus or minus sign indicate a constant. Sensor events are cumulative meaning the speed should be calculated based on the previous speed value. All requirements below must be followed when processing a list of events.

#### Requirements

1. Set Initial speed to 20 kph.
2. For every valid sensor event output the desired speed.
3. Vehicle speed cannot fall below 10 kph.
4. Traffic Clear cannot be applied if Traffic doesn't exist currently.
5. Weather Clear cannot be applied if Weather Rainy doesn't exist currently.
6. Slippery Road Clear cannot be applied if Slippery Road doesn't exist currently.
7. Emergency Turbo cannot be applied if it is Slippery Road.
8. Emergency Turbo cannot be applied more than once.
9. A New Speed Limit Sign clears Emergency Turbo.
10. Any event number  $\geq 10$  is a Speed limit sign.
11. Any event number  $> 100$  is invalid.
12. An event should be ignored if its already applied. Once cleared, an event can be applied again.

- For example, assuming Normal mode, events 50, 1, 3, 1, 2,1 would result in 35 kph. The second Traffic event was ignored but third Traffic event was applied as it was cleared.
- 13. Valid events that cannot be applied because of the above requirements should be ignored.
- 14. Print 'invalid' on invalid events and continue to process.

### Examples

Scenario: Success	Scenario: Ignore	Scenario: Invalid
\$ autonomousDriving NORMAL  Welcome. Driving mode is NORMAL 20 Enter Event: 50 50 Enter Event: 1 40 Enter Event: 2 50 Enter Event:	\$ autonomousDriving SPORT  Welcome. <b>Driving</b> mode is SPORT 20 Enter Event: 7 50 Enter Event: 7 50 Enter Event:	\$ autonomousDriving SAFE  Welcome. <b>Driving</b> mode is SAFE 20 Enter Event: 60 55 Enter Event: 200 Invalid Enter Event:

### Solution

Please submit a command-line solution preferably in Java, although C# or another object-oriented language is acceptable. Upload all source code, tests, executable program and a Readme file with instructions to run the program to the link provided.

If you have not received a link, or have any other questions, please contact [i\\_ext\\_cvs\\_ss\\_interview\\_support@here.com](mailto:i_ext_cvs_ss_interview_support@here.com).

### What we are looking for

Keep in mind that your code should be of production quality and must be maintained by your peers. Solution should demonstrate the following:

- A compiling and working solution
- All requirements and events processed correctly
- Object oriented design
- Unit and integration testing
- Design patterns that follow SOLID principles
- Legible code
- Maintainability
- Extensibility to add new sensor events and driving modes