C++ OOP – Exam Retake (11 September 2022)

Write C++ code for solving the tasks on the following pages.

Code should compile under the C++11 standard.

Submit your solutions here: https://judge.softuni.org/Contests/3598/CPlusPlus-OOP-Retake-Exam-11-September-2022

Only source code will be accepted as solution for each task.

2. Vehicles

You are about to take place in a car race. Wohoo!

Who is better? The man or the machine? Find out in an epic duel between automatic and manual gear transmission vehicles.
Your task is to implement the missing functionality.

The race is played on a RaceTrack, where 2 cars race against one another.
Each of those cars could either be with manual shifts or with automatic ones.

The race is played in **rounds**. Each round you are given a command to execute.
After you execute the command you should print to the standard output the outcome of the command.
The possible commands are:

```
enum class CommandType {
  CREATE_CAR,
  INCREASE_SPEED,
  DECREASE_SPEED,
  ADVANCE
};
```

- ✓ `CREATE_CAR` – creates a car of the specified type (manual or automatic)
- ✓ `INCREASE_SPEED` – increases the speed of the car and consumes fuel. (No distance is travelled)
- ✓ `DECREASE_SPEED` – decreases the speed of the car
- ✓ `ADVANCE` – travels distance based on the current speed of the car

General race rules:

- Both types of cars have a maximum speed and a starts the race with a given quantity of fuel.
- The beginning of the race stars by creating the 2 cars (they can be both manual, both automatic or 1 manual and 1 automatic)
- On each next round a command `INCREASE_SPEED/DECREASE_SPEED/ADVANCE` is received.
- If the `INCREASE_SPEED` step the car **fuel runs out** – **that race is stopped** and the other car is pronounced a winner. The two cars are guaranteed to not run out of fuel at the same time.
- If both cars still have fuel left when the race is over (when there are no more commands) – the car with the biggest traveled distance is announced a winner. If the two cars have the same distance travelled – no winner is announced (Check the output section for concrete examples)

Automatic car rules:

- ✓ When `DECREASE_SPEED` is received – the automatic car doubles that deceleration.
- ✓ Example: current car speed is 80. `DECREASE_SPEED(20)` is received. The car speed should become 80 – 20 * 2 = 40.

Manual car rules:

- ✓ The manual car has a concept of gears/speed shifts. The gears 1-2-3-4-5-6 taken from a real-life example. The reverse shift is skipped for simplicity.
- ✓ Those gears are represented by a vector of integers. The size of the vector is an indicator how many shifts will the manual car have.
- ✓ The values of that vector represent at what speed should the car change from one shift to another.
- ✓ Example: car has maximum speed of 100 and gears/speed shifts { 20, 80 }. This means that the car has 3 shifts – [0, 20), [20, 80), [80, 100]
- ✓ When the car changes from a lower to an upper shift in an `INCREASE_SPEED` command – **the fuel consumption is doubled** for that round.
- ✓ When a car changes from upper to lower shift in a `DECREASE_SPEED` command – nothing special happens
- ✓ A car can "jump" several shifts for a single increase `INCREASE_SPEED` command
- ✓ Example: car has a current speed of 0, maximum speed of 100 and gears/speed shifts { 20, 80 }.
- ✓ `INCREASE_SPEED(90)` is received. The car jumps from shift 0 to shift 2, because it is above 80 speed. The fuel consumption is doubled as normal even though the shift at speed 20 was completely skipped.

Input

The input is already handled for you. You don't need to parse anything additonally

Output

Your RaceTrack should implement a "printInfo()" method. On that method you should print data for the two cars in the following format:

`"Car with index: I has totalDistance: T, is running with S speed, has F fuel left\n"`
Where:
I – index of the car
T – total distance travelled by the car
S – current speed of the car
F – current fuel of the car

You should also implement a "printWinner()" method where you should print one of the following 3 lines:
Car with index: 0 won!

Car with index: 1 won!

`No winner`

See the “Examples” section for better clarity.

Restrictions

Car fuel cannot be negative – 0 is the minimum

Car speed cannot be negative – 0 is the minimum

The car speed cannot exceed its maximum speed

Time limit: 500ms (0.50s)
Memory limit: 16 MB

Examples

Input	Output
6 0 0 100 100 0 0 110 90 1 50 40 3 2 20 3	Starting turn: 1 Car with index: 0 has totalDistance: 0, is running with 50 speed, has 60 fuel left Car with index: 1 has totalDistance: 0, is running with 50 speed, has 50 fuel left Starting turn: 2 Car with index: 0 has totalDistance: 50, is running with 50 speed, has 60 fuel left Car with index: 1 has totalDistance: 50, is running with 50 speed, has 50 fuel left Starting turn: 3 Car with index: 0 has totalDistance: 50, is running with 10 speed, has 60 fuel left Car with index: 1 has totalDistance: 50, is running with 10 speed, has 50 fuel left Starting turn: 4 Car with index: 0 has totalDistance: 60, is running with 10 speed, has 60 fuel left Car with index: 1 has totalDistance: 60, is running with 10 speed, has 50 fuel left No winner
6 0 1 100 100 50 100 0 1 110 50 20 80 1 40 20 3 1 50 30 3	Starting turn: 1 Car with index: 0 has totalDistance: 0, is running with 40 speed, has 80 fuel left Car with index: 1 has totalDistance: 0, is running with 40 speed, has 10 fuel left Starting turn: 2

	Car with index: 0 has totalDistance: 40, is running with 40 speed, has 80 fuel left Car with index: 1 has totalDistance: 40, is running with 40 speed, has 10 fuel left Starting turn: 3 Car with index: 0 has totalDistance: 40, is running with 90 speed, has 20 fuel left Car with index: 1 has totalDistance: 40, is running with 0 speed, has 0 fuel left Car with index: 0 won!
10 0 1 150 120 50 100 0 0 200 90 1 40 20 3 1 50 30 3 2 30 3 1 80 30 3	Starting turn: 1 Car with index: 0 has totalDistance: 0, is running with 40 speed, has 100 fuel left Car with index: 1 has totalDistance: 0, is running with 40 speed, has 70 fuel left Starting turn: 2 Car with index: 0 has totalDistance: 40, is running with 40 speed, has 100 fuel left Car with index: 1 has totalDistance: 40, is running with 40 speed, has 70 fuel left Starting turn: 3 Car with index: 0 has totalDistance: 40, is running with 90 speed, has 40 fuel left Car with index: 1 has totalDistance: 40, is running with 90 speed, has 40 fuel left Starting turn: 4 Car with index: 0 has totalDistance: 130, is running with 90 speed, has 40 fuel left Car with index: 1 has totalDistance: 130, is running with 90 speed, has 40 fuel left Starting turn: 5 Car with index: 0 has totalDistance: 130, is running with 60 speed, has 40 fuel left Car with index: 1 has totalDistance: 130, is running with 30 speed, has 40 fuel left Starting turn: 6 Car with index: 0 has totalDistance: 190, is running with 60 speed, has 40 fuel left Car with index: 1 has totalDistance: 160, is running with 30 speed, has 40 fuel left Starting turn: 7 Car with index: 0 has totalDistance: 190, is running with 140 speed, has 10 fuel left

	Car with index: 1 has totalDistance: 160, is running with 110 speed, has 10 fuel left Starting turn: 8 Car with index: 0 has totalDistance: 330, is running with 140 speed, has 10 fuel left Car with index: 1 has totalDistance: 270, is running with 110 speed, has 10 fuel left Car with index: 0 won!