

# Race Simulation

Проект по ООП

Изготвил: *Димо Бойчев*

*Компютърни науки*

**ФН: 81176**

# Описание

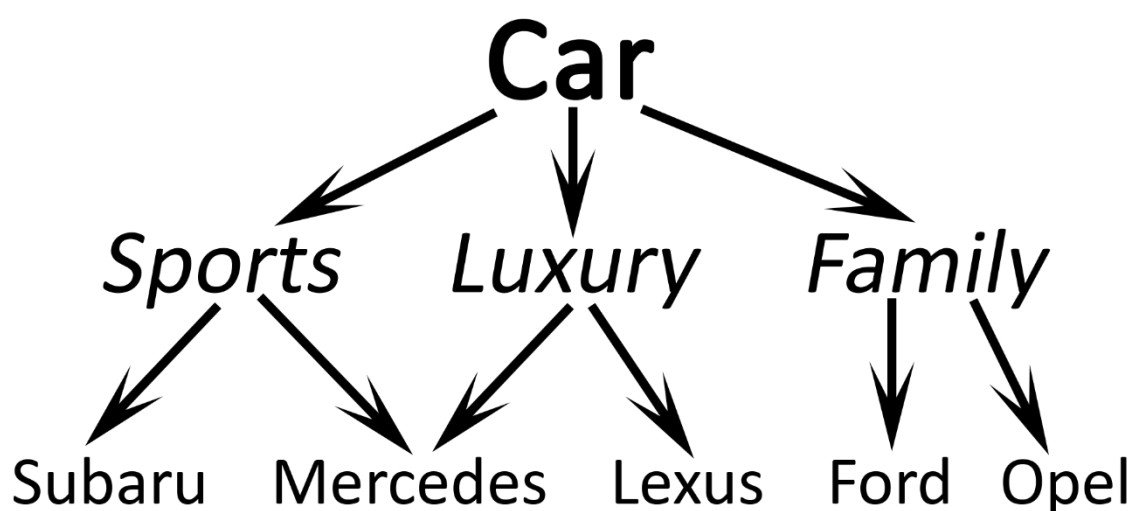
Проектът позволява създаване на **Коли/Пилоти** със собствени параметри, създаване на **Писта** за състезание и възможност за симулация на две от създадените коли върху пистата.

## Класове

- **TRACK**
- **Car** (базов клас)
- *Sports* (наследствен на **Car**)
- *Luxury* (наследствен на **Car**)
- *Family* (наследствен на **Car**)
- Subaru (наследствен на *Sports*)
- Mercedes (наследствен на *Sports* и *Luxury*)
- Lexus (наследствен на *Luxury*)
- Ford (наследствен на *Family*)
- Opel (наследствен на *Family*)

# Иерархия

TRACK



# Полета

## Car

- **char\* name** – име на колата, която създаваш;
- **int hp** – мощността на ѝ в конски сили;
- **int seats** – брой седящи места в нея;
- **int maxSpeed** – максималната скорост, която може да развие;
- **int safety** – ниво на сигурност при завои;
- **double speed** – моментна скорост.

## Sports, Luxury, Family

- КОНСТАНТНИ СТОЙНОСТИ, КОИТО СИ ИЗПОЛЗВАТ ПРИ ПОЛУЧАВАНЕ НА ЕВЕНТУАЛЕН БОНУС ПО ВРЕМЕ НА СЪСТЕЗАНИЕ:
- **static double bonus\_hp** – Sports;
- **static double bonus\_maxSpeed** - Luxury;
- **static double bonus\_safety** - Family;

# Track

- **char\* name** – име на пистата, която създаваш;
- **int length** – дължината ѝ в метри;
- **int laps** – броя обиколки, които да се правят на нея.

# Методи

## Car

- **set** и **get** методи на всички полета;
- **void print() const** - метод, който показва пълната информация за колата;
- **double acceleration() const** – намира ускорението на колата по формулата:  
(мощност в к. с. / брой места в колата / 25)
- **double cur\_speed (int second)** –  
изчислява скоростта, до която колата ще ускори за брой секунди, като взима предвид началната ѝ скорост;
- **void up\_speed (double second)** –  
своевременно актуализира скоростта на колата за секундите, в които ускорява;
- **double meters\_passed (int second)** –  
намира разстоянието в метри, което колата ще измине за определено време в секунди;

- **double lap (int meters)** – намира времето в секунди, за което колата ще измине една обиколка (разстояние в метри);
- **virtual void bonus ()** – виртуална функция, която е имплементирана в подкласовете на **Car** и е различна за всеки от тях;
- **void turn (int meters)** – функцията се извиква, когато колата е в края на обиколката и трябва да премине през завоя, който я извежда отново до началото на пистата;
- **void slow\_down (double seconds) const** – функцията се активира на случаен принцип, т. е. взима завоя по лош начин;
- **void time (double seconds) const** – помощна функция, която принтира секундите във формат мин:сек, ако секундите са повече от 60;

- **virtual void race (Car& c, int length, int laps)** – виртуална функция, която е имплементирана в подкласовете на **Car** – с нея се активира самото състезание между две коли.

## **Sports, Luxury, Family**

- **void bonus ()** – функция, която се активира на произволен принцип в края на всяка обиколка – колата получава бонус към някой от стойностите си;
- **void race (Car& c, int length, int laps)** – функцията симулира състезание и приема като параметри противниковата кола, дължината на пистата в метри и обиколките на пистата;
- **double get\_bonus\_hp () const, update\_hp ()** – Sports
- **double get\_bonus\_maxSpeed () const, update\_maxSpeed ()** – Luxury



- **double get\_bonus\_safety () const, update\_safety ()** – Family
- Функциите активират бонуса на колата, спрямо категорията ѝ и актуализират параметрите ѝ;
- **\*Забележка:** Тъй като класът Mercedes е наследник и на Sports, и на Luxury, в него също са имплементирани методите **bonus ()** и **race (Car& c, int length, int laps)**.

## Track

- **set** и **get** методи на всички полета;
- **void RACE (Car& A, Car& B)** – метод, който показва информация за пистата и симулира състезание между колите **A** и **B**.

# Тестване

*Track A (“име на пистата”, дължина в метри, обиколки);*

*Car B (“име на колата”, мощност, максимална скорост);*

*Car C (“име на колата”, мощност, максимална скорост);*

**A.RACE(B, C);**