

# ООП - Практикум

## Група 2

### Задача 1

Да се дефинира абстрактен клас **Device**, който поддържа следните данни:

- Модел – символен низ
- Марка – символен низ
- Цена – цяло число

Да се дефинират методи за принтиране на информацията на едно устройство, писане и четене във файл.

Нека класът **Phone** наследи **Device** като го надгражда със следните характеристики:

- Брой камери - цяло число
- SecurityLockType – избор между лицево разпознаване и пръстов отпечатък

Нека класът **Laptop** също наследи **Device** като за него има следната информация:

- Операционна система – MacOS, Linux, Windows
- Дали е с USB портове или с Thunderbolt
- Брой портове-цяло число

Нека **Phone** и **Laptop** имат свои имплементации на методите за принтиране на информацията, писането и четенето от файл.

Да се реализира клас **TechnoShop**, който представлява магазин за техника. Нека той има име и съдържа списък от устройства. Да се напишат методи за добавяне и премахване на устройство, премахването да става по подадени модел и марка, записване на информацията на всички устройства във файл и четене на информация за магазин и неговите артикули от файл.

Помислете какво още ще ви е нужно, за да реализирате класа **TechnoShop**. Когато пишете във файл записвайте на отделен ред всяко устройство и неговия вид, за да бъде лесно четим файла след това. При четене от файл, ако програмата попадне на непозната фигура да хвърля грешка.

### Задача за реализация на собствен String клас

Да се създаде клас **MyString** и нека той поддържа следната функционалност:

- Всички видове конструктори и деструктор

- Метод **append**, който залепя в края на стринга друг стринг или символ
- Оператор за достъп на символ(констатна и неконстантна версия)
- Метод `length()`, който връща големината на стринга
- Оператор `=`
- Операторите `==` и `!=`
- Оператори за четене и писане от конзолата

## Бонус

Да се напише функция **SplitBy(const char symbol)**, която разделя стринга по подадения символ и връща вектор от разделените части.

## Задача за реализация на собствен Vector клас

Да се създаде клас **MyVector** за тип `int` и нека той поддържа следната функционалност:

- Всички видове конструктори и деструктор
- Метод **add** за добавяне на елемент
- Оператор за достъп на елемент от вектора(константна и неконстатна версия)
- Оператор `=`
- Оператор `+` и `+=`