

3. Домашна работа

1 Queue*

Създайте клас **Queue**. С методи **Enqueue**(**int** element) и **Dequeue**(). Представете си опашка пред магазина. Хората се нареждат един след друг и който е на ред трябва да бъде обслужен. Ако нов човек дойде той не може да прережда другите хора и застава на края на опашката. Целта на метода Enqueue е всеки път да добавя елемента най – отзад на опашката, а метода Dequeue трябва да връща текущия елемент който трябва да бъде обработен (първия на опашката). Можете да прочетете за Queue още в интернет, като целта ви е да направите такъв клас.

2 Горива*

Създайте клас **Fuel** със свойството **Price**: **decimal** и **Type**: **string**. Създайте клас **GasStation** със свойството **Fuels**: **List<Fuel>** и **Name**: **string**. Създайте списък от бензиностанции и добавете по няколко горива в тях.

Създайте метод **LowestPrice**(**List<GasStation>** stations): **GasStation**, който връща бензиностанцията с най – ниските като цена горива (средното).

Към класовете добавете подходящи конструктори.

3 Песен

Създайте клас **Song** със свойствата **Duration**: **double**, **Name**: **string**. Създайте клас **Author** със свойствата **Name**: **string** и **Songs**: **List<Song>**. Създайте списък от автори и добавете по няколко песни към тях. Програмата отпечатва автора и всички негови песни.

Към класовете добавете подходящи конструктори.

4 Worker

Създайте клас **Worker** със свойствата **Name**: **string**, **HourlyRate**: **decimal** и метода **YearlySalary**(): **decimal**. Метода **YearlySalary** намира годишната заплата на работника.

Приемаме, че един месец има 22 работни дни и по 8 часа на ден, а свойството **HourlyRate** е заплатата на работника за час. Към класа добавете и подходящ конструктор.