**Задание 1.** [CSV](https://ru.wikipedia.org/wiki/CSV)-файл data.csv содержит информацию об отпусках сотрудников некоторой компании. Каждая строка – это имя сотрудника, дата начала отпуска, дата окончания отпуска. Вот фрагмент файла:

Dlsw Fmbnaswsg,1/25/2016,2/3/2016

Iqbidtaw Ndxqjdpsg,6/8/2015,6/19/2015

Lrnywj Fdgyzw,6/8/2015,6/16/2015

Имена сотрудников зашифрованы. Полный файл – в приложении к заданию.

1. Прочитайте содержимое файла в коллекцию объектов в памяти. Если какая-то строка файла содержит ошибку, то эта строка игнорируется, информация об ошибке и номере строки выводится на консоль, но чтение файла не прекращается.
2. Найдите среднюю продолжительность отпуска в компании.
3. Найдите среднюю продолжительность отпуска для каждого сотрудника. (выведите на консоль пары «имя сотрудника – средняя продолжительность отпуска»).
4. Сохраните информацию о средней продолжительности отпуска сотрудников в JSON-файл persons.json. Желаемый вид итогового JSON-файла поясняет фрагмент ниже.

[

{

"name": "Dtld Fmdwqntxpdjd",

"duration": 4.4

},

{

"name": "Rnrzw Qnaswjp",

"duration": 3.5

},

. . .

{

"name": "Tdqbwnd Qdasgtnpsgd",

"duration": 9

}

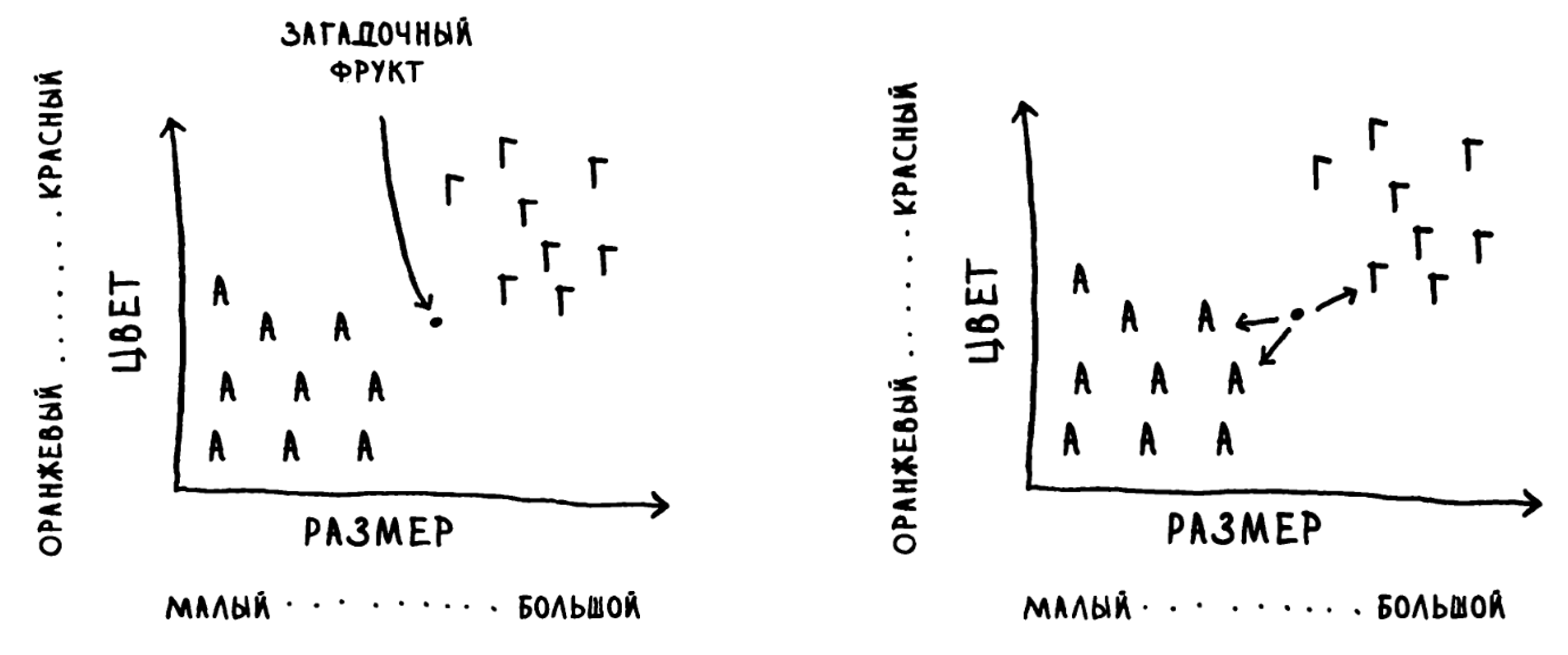
]

**Внимание:** пункты 2–3 желательно закодировать, используя возможности PLINQ или класса Parallel.

**Задание 2.** *Метод k-ближайших соседей* – один из простейших алгоритмов машинного обучения (<https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm>).

Суть алгоритма: пусть имеется набор объектов с известными **названиями** и числовыми **характеристиками**. Эти характеристики дают возможность описать объект как точку в ***n***-мерном пространстве. Пусть есть объект, у которого заданы характеристики, но нет названия. Чтобы найти название этого объекта, найдём ***k*** ближайших к нему известных объектов. То название, которое будет у ближайших соседей повторяться чаще, сделаем названием нашего неизвестного объекта (алгоритм классификации).

Пример: апельсины и грейпфруты, ***k=3***. Неизвестный объект оказался апельсином.



1. Необходимо реализовать алгоритм метода k-ближайших соседей в двумерном пространстве характеристик. Вход алгоритма: набор известных объектов, две характеристики неизвестного объекта и значение ***k***. Каждый объект состоит из строкового названия и двух (для простоты) числовых характеристик (можно использовать кортежи). Выход алгоритма: строка с названием неизвестного объекта. Постарайтесь при реализации алгоритма использовать возможности класса Parallel или технологии PLINQ.
2. Создайте консольное приложение для тестирования алгоритма. При старте приложение читает входные данные из текстового файла (сам файл вам предоставляется). Затем приложение использует генератор данных в виде **асинхронного потока**, который раз в секунду выдаёт характеристики неизвестного объекта – этот объект классифицируется алгоритмом. Протестировать 10 неизвестных объектов и вывести их характеристики и названия на консоль.