Домашнее задание №2

## Задача 1.

Необходимо создать консольную утилиту для обработки файлов, которые представляют собой текст некоторых литературных произведений. Утилита должна вычислять частоту встречаемости слов в файле и выводить на консоль самые часто встречающиеся слова. Регистр слов не различать (например, *The* и *the* рассматривать как одно слово).

Утилита должна использовать следующие флаги командной строки:

* -file или -f – этот флаг служит для указания имени текстового файла. Флаг обязательный. Если он не задан или указанный файл не существует, утилита прекращает свою работу.
* -top или -t – флаг используется, чтобы задать число выводимых слов. Если флаг не задан, считается, что надо вывести 10 слов.

## Задача 2.

Разработайте консольное приложения для управления списком задач – **TODO App**. Работа с приложением выполняется при помощи флагов командной строки:

* -list – при наличии этого булевого флага приложение выводит на консоль все задачи. Каждая задача включает порядковый номер, название, флаг завершения, дату создания и дату завершения.
* -complete=num – этот флаг сообщает, что задачу с порядковым номером num надо сделать завершённой.
* -task="some task"– при помощи этого флага создаётся новая незавершённая задача.

Приложение TODO App должно хранить данные о задачах в JSON-файле.

## Задача 3.

Необходимо создать в отдельном пакете и затем протестировать в консольном приложении структуру для представления кеша произвольных объектов в памяти. Для доступа к объекту в кеше используется строковый ключ. Кеш имеет ограниченную ёмкость и механизм для автоматического удаления из кеша «старых» элементов.

* Функция создания кеша должна принимать в качестве аргумента положительное целое число. Это ёмкость кеша, то есть максимальное количество элементов в кеше.
* Снабдите кеш методом Set(key string, obj any) для добавления объекта в кеш. Если в кеше уже есть объект с ключом key, то этот старый объект заменяется на obj (с обновлением времени последнего доступа). Если количество элементов в кеше равно его ёмкости, то перед добавлением нового объекта из кеша удаляется элемент, к которому дольше всего не обращались.
* Снабдите кеш методом Get(key string) any для получения объекта по ключу. Если в кеше нет объекта с ключом key, метод возвращает значение nil. Если в кеше есть объект с ключом key, метод возвращает этот объект, при этом время последнего доступа к элементу кеша обновляется.
* Создайте метод Remove(key string) bool для удаления из кеша объекта по ключу. Если объект нашли по ключу key и удалили, метод возвращает значение true. Если в кеше нет объекта с ключом key, метод возвращает значение false.
* После создания, кеш должен автоматически сканировать свои элементы (раз в секунду) и удалять те элементы, к которым не обращались более 10 секунд.