

## Задание по «Вычислительной практике» 2017-2018 гг. студента Баранова А. А.

**Задание:** Реализовать предсказатель, который зачитывал бы заранее подготовленные данные результатов футбольных матчей, и выводил бы предполагаемый исход (победа одной из команд или ничья). Github: [https://github.com/NoruNoruBim/Summer\\_practice\\_2018](https://github.com/NoruNoruBim/Summer_practice_2018)

Дата сдачи	Что нужно сделать	Пояснения
10 июня 2018 года	<ol style="list-style-type: none"><li>1. Завести репозиторий на <i>github</i>;</li><li>2. Прочитать литературу по нейронным сетям и машинному обучению;</li><li>3. Почитать про <i>unittest</i> для <i>Python</i> (или его аналог, если используете <i>C++</i>) и написать свой первый тест, используя <i>mock</i>. На этом этапе можно определить интерфейс будущего класса <i>Requester</i>;</li><li>4. Ознакомиться с методами парсинга данных из <i>html</i>;</li><li>5. Найти спортивный сайт, откуда можно скачать результаты футбольных матчей. Написать класс <i>Requester</i>, использующий модуль <i>requester</i> или его аналог. Обкачать одну страницу, распарсить её и проверить выходные данные;</li><li>6. Подготовить план по реализации задания;</li></ol>	Соответственно в тестах обкачивать на самом деле ничего не нужно. Для этого и нужен <i>mock</i> — класс, который будет иметь интерфейс схожий с настоящим <i>Requester</i> ’ом, но возвращающий <i>html</i> заранее скаченной страницы. Не стоит забывать, что если выбранный сайт имеет динамический контент, то нужно использовать <i>Selenium</i> , который эмулирует работу браузера. В тестах надо проверить валидность парсера этой <i>html</i> .
14 июня 2018 года	<ol style="list-style-type: none"><li>1. Оформить парсер в виде класса <i>Requester</i>;</li><li>2. Написать для него тест;</li><li>3. Подготовить файл с командами и результатами;</li><li>4. Написать предсказатель с помощью нейронных сетей;</li></ol>	