1. API должен содержать методы:
   1. Получить множество задач
   2. Отправить решение на проверку
   3. Отправить множество решений на проверку
   4. Перепроверить решение
   5. Перепроверить множество решений
      1. Описание: метод должен переотправлять все заданные решения, которые уже имеются в базе, заново на проверку.
      2. Входные данные:
         1. Список идентификатором решений
            1. Тип: список целых чисел
      3. Выходные данные:
         1. Список идентификаторов результатов
            1. Тип: список целых чисел
   6. Получить результат решения
      1. Описание: метод должен возвращать результат решения по заданному идентификатору.
      2. Входные данные:
         1. Идентификатор решения
            1. Тип: целое число
      3. Выходные данные:
         1. Результат решения
            1. Тип: класс SubmitResult
   7. Получить множество результатов
      1. Описание: метод должен соответствующий возвращать список результатов по заданному списку идентификаторов этих результатов.
      2. Входные данные:
         1. Список идентификаторов результатов:
            1. Тип: список целых чисел
      3. Выходные данные
         1. Список результатов решений
            1. Тип: список объектов типа SubmitResult
   8. Получить ссылку на условие задачи
      1. Описание: метод должен возвращать ссылку на условие задачи по заданному идентификатору контеста и идентификатору задачи в контесте.
      2. Входные данные:
         1. Идентификатор контеста
            1. Тип: целое число
         2. Идентификатор задачи в контесте
            1. Тип: строковое значение
      3. Выходные данные:
         1. Ссылка на условие задачи
            1. Тип: строковое значение
   9. Получить список поддерживаемых языков программирования
      1. Описание: метод должен возвращать список языков программирования
      2. Входных данных нет.
      3. Выходные данные:
         1. Список языков программирования
            1. Тип: список строковых объектов
2. Интерфейс спецификации API
   1. Интерфейс спецификации должен генерироваться с помощью готового инструмента Swagger.
   2. Интерфейс должен включать разделы описания каждого контроллера:
      1. Список методов, предоставляемых контроллером.