

Задание 4

Требуется разработать веб-приложение для решения задачи оптимизации генетическим алгоритмом. Приложение состоит из HTTP-сервиса (серверная часть) и страницы HTML5 (клиентская часть). Серверная часть приложения использует компонент из первого задания для решения задачи оптимизации.

Пользовательский интерфейс позволяет задать параметры задачи. Для задачи коммивояжера отображается матрица расстояний между городами. Также пользовательский интерфейс предоставляет кнопки для запуска и остановки процесса оптимизации. В ходе поиска решения отображается номер итерации, лучший экземпляр и его метрика.

Программный интерфейс серверной части зависит от варианта.

Вариант А. Серверная часть поддерживает два запроса:

1. `GET /initial?...` для формирования начальной популяции. Параметры задачи передаются в строке запроса. Ответом является JSON-документ с параметрами задачи и начальной популяцией.
2. `POST /next` для выполнения очередного шага эволюции. Параметры задачи и исходная популяция передаются в теле запроса. В ответ возвращается документ JSON с новой популяцией, лучшим экземпляром и его метрикой.

Вариант Б. Серверная часть поддерживает три запроса:

1. `PUT /experiments` для создания на сервере нового эксперимента. Параметры задачи передаются в теле запроса. В ответ возвращается идентификатор эксперимента.
2. `POST /experiments/id` для выполнения очередного шага в указанном эксперименте. В ответ возвращается лучший экземпляр и его метрика.
3. `DELETE /experiments/id` для удаления эксперимента с указанным идентификатором.

Общие требования

В решение должны быть включены сценарные тесты для серверной части. Сценарные тесты описаны здесь:

<https://timdeschryver.dev/blog/how-to-test-your-csharp-web-api#a-simple-test-using-xunit-and-39-s-fixtures>