

Веб интерпретатор одномерной и двумерной машин Тьюринга

Итерация 3

КУРАТОР:	МАКСИМ ДОБРОХВАЛОВ
СТУДЕНТ-МАГИСТР:	КИРИЛЛ ВИНОГРАДОВ
СТУДЕНТЫ-БАКАЛАВРЫ:	АНТОН КИБАРДИН
	ДМИТРИЙ ЧЕШУИН
	КИРИЛЛ КРЫЖАНОВСКИЙ

Задача и функциональность

Задача: создать веб-приложение, которое будет интерпретатором одно- и двумерной машины Тьюринга, а также движком для решения простейших задач на ней.

Функциональность:

- создание, редактирование, отображение, импорт и экспорт программ для Машин,
- визуальная симуляция работы Машин,
- выполнение программы по шагам,
- примитивные инструменты отладки,
- общие интерфейсы для создания задач,
- задачи-примеры из <https://github.com/OSLL/adfmp20-turing> и курса Информатика,
- сохранение результатов решения задач в БД,
- поддержка пользователей,
- поддержка протокола LTI*.

Инструкция по развертыванию и запуску проекта

Клиент:

- Загрузить проект https://github.com/moevm/mse_turing_tasks/tree/master/client
- В папке проекта установить зависимости с помощью команды **npm install**
- Для запуска выполнить команду **npm run serve**

Сервер:

- Установить интерпретатор Python3
- Загрузить проект https://github.com/moevm/mse_turing_tasks/tree/master/server
- Установить зависимости с помощью команды: **pip install -r requirements.txt**
- Создать новый исполняемый файл *.py и добавить в него строки:

```
from app import app  
app.run()
```

- Запустить созданный ранее файл

БД:

- Установить зависимости командой **pip3 install -r requirements.txt**
- Выполнить скрипт **startMongo.sh**

Задачи 3 итерации

- Добавить в машину обработку внутренних ошибок: бесконечный цикл, вызов несуществующего состояния, вызов несуществующего перемещения - **сделано**
- Добавить возможность выполнения программы по шагам с клиента – **не сделано**
- Добавить отрисовку поля для 1 и 2 мерных машин - **сделано**
- Добавить страницу регистрации - **сделано**
- Добавить сессии пользователей - **сделано**
- Осуществить переход на MongoDB Atlas - **сделано**
- Добавить сохранение и загрузку машины из файла
- Добавить изменение полей таблицы состояний и начальной координаты - **сделано**
- Улучшить дизайн UI: корректировка размера и позиции элементов, корректировка цветовой схемы и т.д. - **сделано**
- Улучшить логгирование: добавление дополнительной информации (время, адрес сервера и т.д.), представление структуры и т.д. – **не сделано**
- Разработать юнит-тесты – **не сделано**
- Разработать докер-контейнер – **не сделано**

Результаты работы: Клиент

1. Подключена авторизация существующего и регистрация нового пользователя на клиенте
2. Добавлена отрисовка ленты и матрицы для одномерной и двумерной машин
3. Добавлена возможность заполнять клетки символами, перемещаться по ленте и плоскости, выбирать начальную позицию для машины

Результаты работы: Сервер - БД

1. Подключена облачная БД mongo atlas для авторизации и регистрации
2. Добавлена простейшая реализация sso авторизации через серверное api: routes - /registration и /login, а так же возможность получать список машин пользователя: route - /session/blueprints
3. Добавлена возможность сохранять, загружать и удалять машины через api(временно на виртуальной базе данных): routes - /session/savebpc, /session/loadbpc и /session/deletebpc
4. Добавлена возможность пошагового выполнения машины через api: routes /session/debug/start и /session/debug/next

Планы на следующую итерацию

- Добавить возможность выполнения программы по шагам с клиента
- Улучшить логгирование: добавление дополнительной информации (время, адрес сервера и т.д.), представление структуры, информация для отладки и т.д.
- Разработать юнит-тесты
- Разработать докер-контейнер
- Добавить работу с файлами
- Добавить визуальное представление шагов
- Перенести сохранение программ из виртуальной БД в основную
- Добавить инструменты отладки
- Добавить программы-примеры