



Санкт-Петербургский государственный университет
Кафедра системного программирования

Разработка настольной версии «Конструктора вычислителей»

Илья Владимирович Муравьев, группа 20.Б11-мм

Научный руководитель: к.т.н. Ю.В.Литвинов, доцент кафедры системного программирования

Санкт-Петербург
2022

- Уровень усвоения материала на курсе «Теория автоматов и формальных языков» «объективно низкий» \implies была начата разработка веб и настольной версий «Конструктора вычислителей»
- Визуализация недетерминированных вычислений малоразвита, но существенна при изучении и отладке недетерминированных вычислителей
- В настольной версии требовалась поддержка сохранения автоматов, отмены и повтора операций

Постановка задачи

Целью работы стало усовершенствование настольной версии «Конструктора вычислителей»

Задачи:

- Произвести обзор используемых аналогами способов отладки недетерминированных вычислителей
- Упростить отладку недетерминированных вычислителей в «Конструкторе вычислителей»
- Поддерживать автоматы с дополнительными свойствами состояний, в частности автомат Мура
- Реализовать сохранение и открытие автоматов
- Реализовать отмену и повтор совершённых действий, согласованные с групповыми операциями
- Обновить пользовательскую документацию
- Провести тестирование и апробацию

- JavaFX (теперь OpenJFX)
- Kotlin
- TornadoFX
- JUnit 5
- MockK
- Gradle
- `kotlinx.serialization` (ещё рассматривались Gson и Jackson)

Рассмотренные аналоги:

- [JFLAP](#) (наиболее полный инструмент согласно «Fifty Years of Automata Simulation: A Review»)
- [Automata-Visualizer](#) (поддержка ДКА, НКА, эpsilon-НКА, ДМП-автомата, МП-автомата и МТ)
- [Automaton Simulator](#) (поддержка ДКА, НКА, эpsilon-НКА, ДМП-автомата и МТ)
- [FSM simulator](#) (поддержка ДКА, НКА и эpsilon-НКА)
- [automatonsimulator.com](#) (поддержка ДКА, НКА, эpsilon-НКА, ДМП-автомата и МП-автомата)

Обзор аналогов (JFLAP)

JFLAP : (even palindrome.jff)

File Input Test View Convert Help

Editor **Simulate: 01**

Traceback

q0	01
Z	
q1	01
Z	
q2	01
Z	

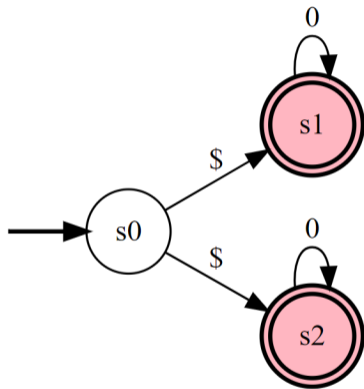
```
1, 0 : 10
1, Z : 1Z
0, Z : 0Z
0, 0 : 00
0, 1 : 01
1, 1 : 11
```

```
graph TD
    q0((q0)) -- "1, 0 : 10" --> q1((q1))
    q0 -- "1, Z : 1Z" --> q1
    q0 -- "0, Z : 0Z" --> q1
    q0 -- "0, 0 : 00" --> q1
    q0 -- "0, 1 : 01" --> q1
    q0 -- "1, 1 : 11" --> q1
    q1 -- "1, 1 : ε" --> q1
    q1 -- "0, 0 : ε" --> q1
    q1 -- "ε, Z : Z" --> q2((q2))
```

q2 01	q1 01	q1 01	
Z	0Z	10Z	

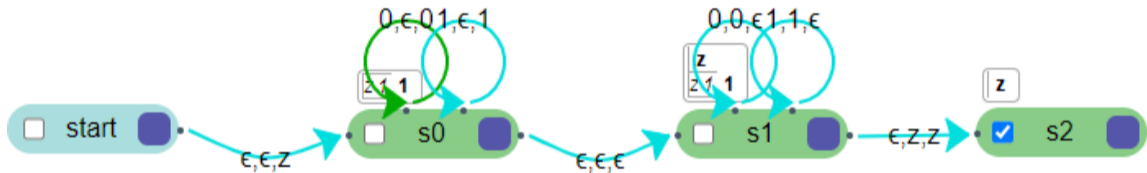
Step Reset Freeze Thaw Trace Remove

Обзор аналогов (Automaton Simulator и FSM simulator)



0001

Обзор аналогов (automatonsimulator.com)



Реализация (отладка недетерминированных вычислителей)

*C:\Users\murav\Documents\automaton-creator\even palindromes.atmtn

File Edit

Nondeterministic pushdown automaton with epsilon transitions

1;z/1z
1;1/11
0;0/00
0;z/0z
1;0/10
0;1/01

Input tape: 0000
Stack: z
Frozen:

Input tape: 0000
Stack: 00z
Frozen:

Input tape: 0000
Stack: 00z
Frozen:

Step	S2 [Rejected]	S1 [Running]	S1 [Running]
Step by state	Input tape: 0000 Stack: z	Input tape: 0000 Stack: z	Input tape: 0000 Stack: 00z
Step by closure			

Реализация (отладка недетерминированных вычислителей)

*C:\Users\murav\Documents\automaton-creator\even palindromes.atmtn

File Edit

Nondeterministic pushdown automaton with epsilon transitions

1;z/1z
1;1/11
0;0/00
0;z/0z
1;0/10
0;1/01

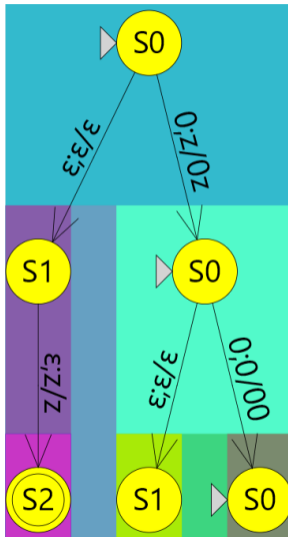
State transition diagram showing states S0, S1, and S2. Transitions are labeled with input/output symbols and stack operations. S0 is the start state.

Stop	▼ S2 [Rejected]	▼ S1 [Running]	▼ S1 [Running]
Input tape	0000	0000	0000
Stack	z	z	00z
Frozen	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

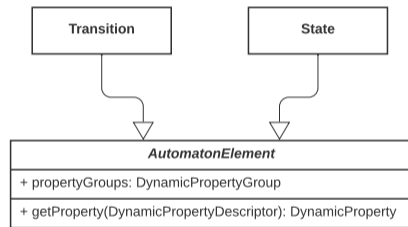
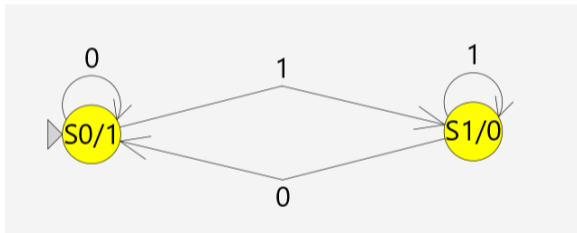
Execution tree diagram showing states S0, S1, and S2. Transitions are labeled with input/output symbols and stack operations. S2 is a rejected state.

Input tape	Stack	Frozen
0000	z	<input type="checkbox"/>

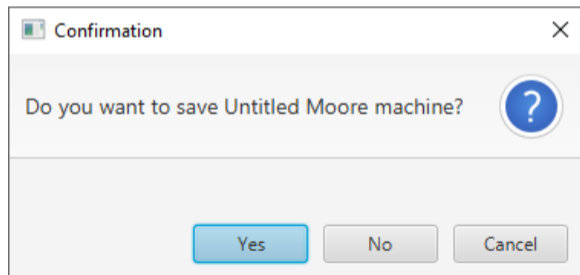
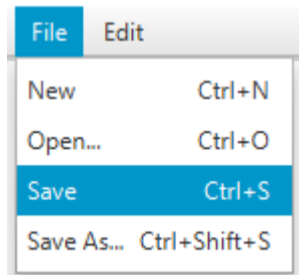
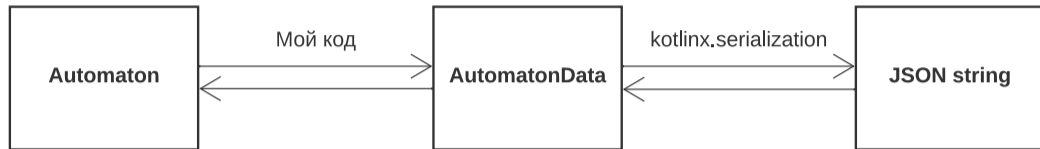
Реализация (отладка недетерминированных вычислителей)



Реализация (автоматы с дополнительными свойствами состояний)



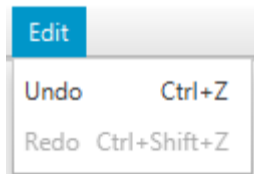
Реализация (сохранение и открытие автоматов)



Реализация (отмена и повтор действий)

Поддерживаемые действия:

- создание и удаление состояний и переходов
- изменение свойств состояний и переходов
- передвижение состояний



- Покрытие модели по веткам и по инструкциям 90%
- Обновлена [пользовательская документация](#)
- Средняя оценка System Usability Scale 75.8 (по результатам шести заполненных опросов)

Заключение

- Произведен обзор используемых аналогами способов отладки недетерминированных вычислителей
- Для упрощения отладки недетерминированных вычислителей добавлены всплывающие подсказки и реализовано представление недетерминированных вычислений в виде интерактивного дерева
- Поддержаны автоматы с дополнительными свойствами состояний, в частности автомат Мура
- Реализованы сохранение и открытие автоматов
- Реализованы отмена и повтор совершённых действий, согласованные с групповыми операциями
- Обновлено пользовательская документация
- Проведены тестирование и апробация

Ознакомиться с реализацией можно в [репозитории](#) на GitHub