

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«Южно-Уральский государственный университет (национальный исследовательский университет)»
Высшая школа электроники и компьютерных наук
Кафедра системного программирования

Разработка языка программирования для Машины Тьюринга

Научный руководитель:
ст. преподаватель кафедры СП
В.В. Варкентин

Автор:
студент группы КЭ-433
М.Н. Масимов

Челябинск 2024

ЦЕЛЬ И ЗАДАЧИ РАБОТЫ



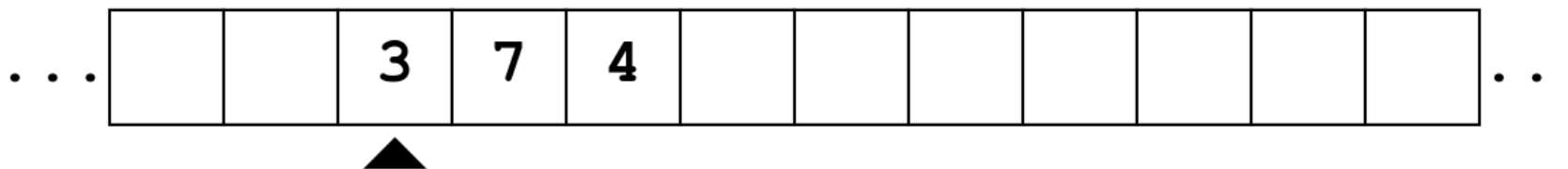
Цель

Разработка языка программирования, позволяющего создавать программы для Машины Тьюринга.

Задачи

1. Спроектировать язык программирования для Машины Тьюринга и разработать соответствующий интерпретатор.
2. Разработать программный интерфейс для эмуляции работы Машины Тьюринга.
3. Реализовать графический интерфейс для эмулятора, позволяющий исполнять программы, написанные на проектируемом языке, и визуализировать работу Машины Тьюринга.

УСТРОЙСТВО МАШИНЫ ТЬЮРИНГА



Бесконечная лента

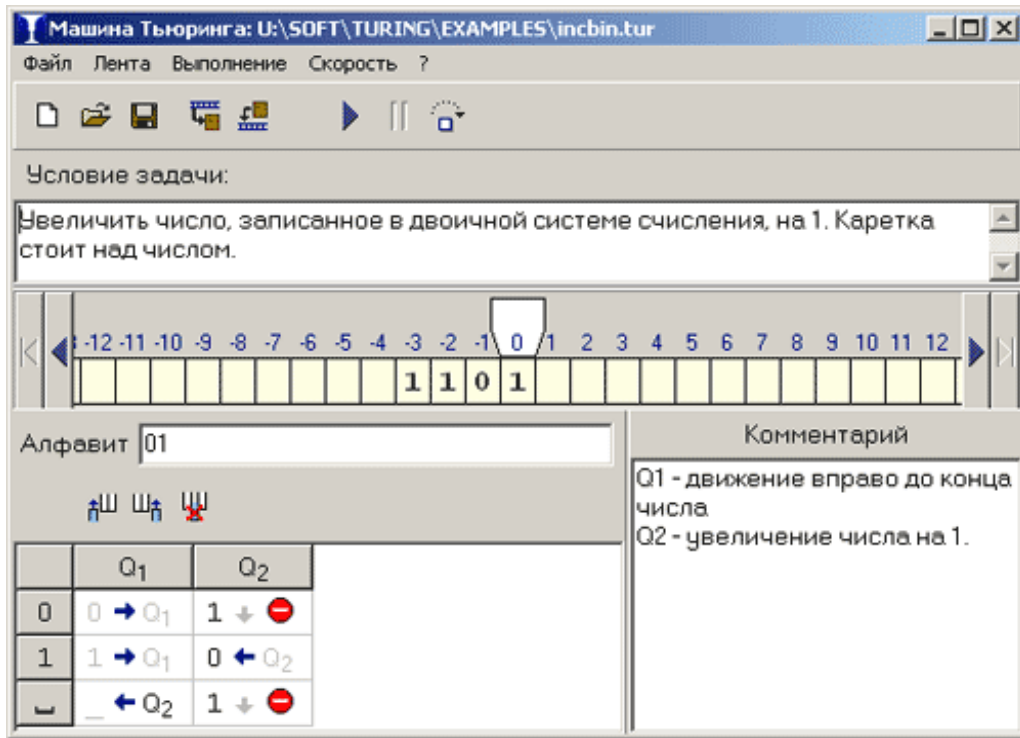
	_	0	1	2	3	4	5	6	7	8	9
q_1	c_0_L	q_10R	q_11R	q_12R	q_13R	q_14R	q_15R	q_16R	q_17R	q_18R	q_19R
c_0	q_0_R	c_00L	c_02L	c_04L	c_06L	c_08L	c_10L	c_12L	c_14L	c_16L	c_18L
c_1	q_01N	c_01L	c_03L	c_05L	c_07L	c_09L	c_11L	c_13L	c_15L	c_17L	c_19L

Таблица переходов

ОБЗОР АНАЛОГОВ

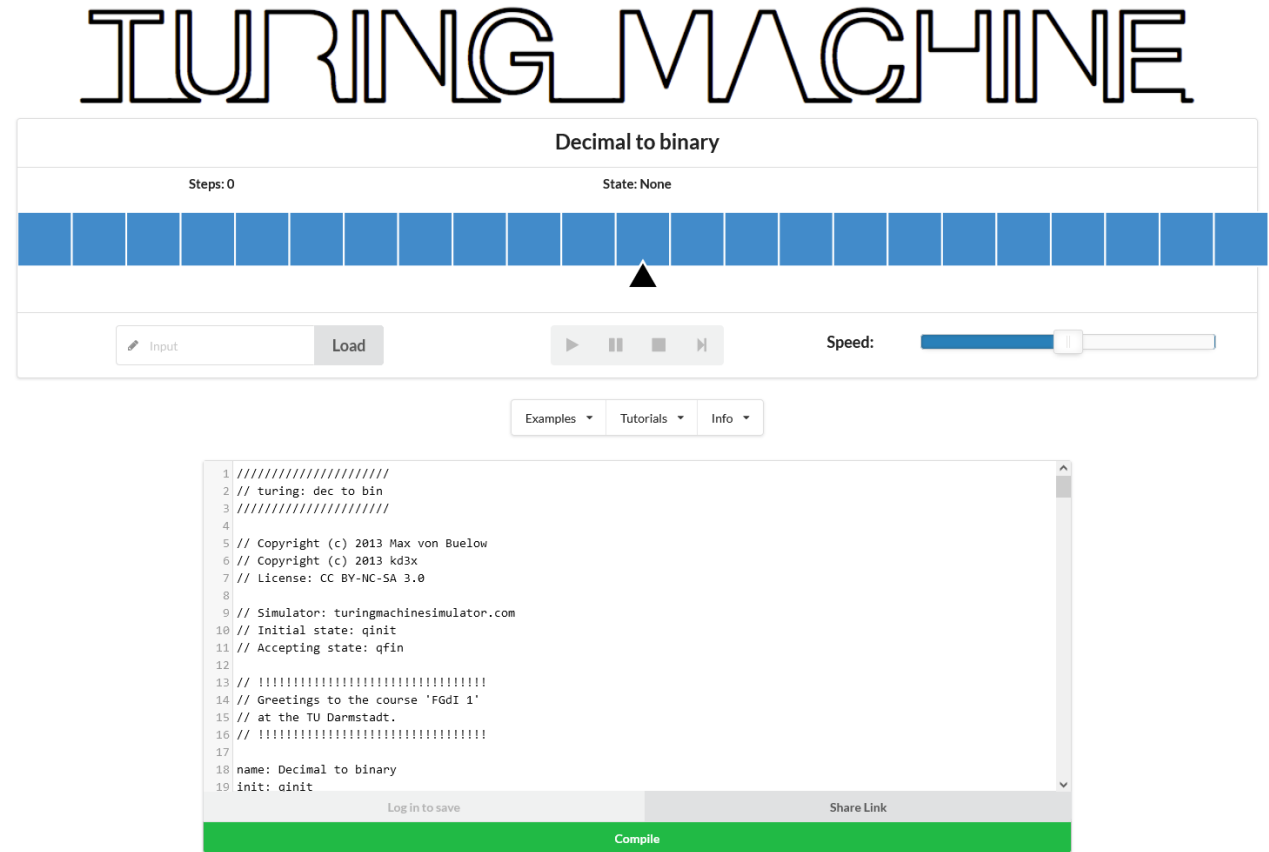


Log In Sign Up



Тренажер «Машина Тьюринга»
К.Ю. Полякова

<https://github.com/0x20cd/turing>



Онлайн-симулятор «Turing Machine»

КОНЦЕПЦИЯ



	_	0	1	2	3	4	5	6	7	8	9
q_1	c_0_L	$q_1\mathbf{0}R$	$q_1\mathbf{1}R$	$q_1\mathbf{2}R$	$q_1\mathbf{3}R$	$q_1\mathbf{4}R$	$q_1\mathbf{5}R$	$q_1\mathbf{6}R$	$q_1\mathbf{7}R$	$q_1\mathbf{8}R$	$q_1\mathbf{9}R$
c_0	q_0_R	$c_0\mathbf{0}L$	$c_0\mathbf{2}L$	$c_0\mathbf{4}L$	$c_0\mathbf{6}L$	$c_0\mathbf{8}L$	$c_1\mathbf{0}L$	$c_1\mathbf{2}L$	$c_1\mathbf{4}L$	$c_1\mathbf{6}L$	$c_1\mathbf{8}L$
c_1	$q_0\mathbf{1}N$	$c_0\mathbf{1}L$	$c_0\mathbf{3}L$	$c_0\mathbf{5}L$	$c_0\mathbf{7}L$	$c_0\mathbf{9}L$	$c_1\mathbf{1}L$	$c_1\mathbf{3}L$	$c_1\mathbf{5}L$	$c_1\mathbf{7}L$	$c_1\mathbf{9}L$



1. Для состояния $q[1]$, для всех десятичных цифр $d[n]$: сдвинуть каретку вправо.
2. Для всех состояний $c[k]$, для всех десятичных цифр $d[i]$:
записать в ленту $d[(2 * i + k) \% 10]$, передвинуть каретку влево,
перейти в состояние $c[i / 5]$.
3. $q_1_ \rightarrow c_0_L$; $c_0_ \rightarrow q_0_R$; $c_1_ \rightarrow q_0\mathbf{1}N$.

СИНТАКСИС ЯЗЫКА



A: <объявл_симв> { , <объявл_симв> } .

Q: <объявл_сост> { , <объявл_сост> } .

{<набор_состояний>:

<набор_символов> -> <символ> , <напр> , <состояние>

{ ; <набор_символов> -> <символ> , <напр> , <состояние> } .

}

СЕМАНТИКА ЯЗЫКА



Конструкция	Примеры
Объявление алфавита	$A: \text{null} = \text{'_'}, a = \text{'X'}, b[0..3] = \text{"ABCD"}, \text{foo}, x[0..1][2..1], x[1][1] = \text{'!'}$.
Объявление состояний	$Q: q[0..1][0..3], q[0][0] = \text{start}, q[1][3] = \text{end}, s1[0..3], \text{bar}$.
Числовые диапазоны	$0..4, 7..-2, n..2*k+3$.
Числовые последовательности	$1..2 \ \& \ 5..4 \ \& \ 3 \ \& \ a..9 \ \& \ a*b..-3$
Арифметические выражения	$5 + 3 * 2^4 \% 7 - (4 + 2 * a^{-(b+c)})$
Строковые литералы	$\text{'test 123'}, \text{"\u{1f600}\u{0d}\u{0a}"}, \text{'"\'}$.
Символьные диапазоны	$\text{'a'..'f'}, \text{"9".."0"}$.
Составные строки	$\text{'te'} \ \& \ \text{"s".."t"} \ \& \ \text{" " } \ \& \ \text{'1'..'3'}$
Ссылки	$a, b[1], \text{foo}, x[1][1], q[1][n*2], s1[3], \text{bar}$.
Итерируемые ссылки	$b\{n\}, b\{x \mid 1..0 \ \& \ 3\}, q\{x\}\{y\}, q\{_ \}[2], q[1]\{_ \mid 2..1\}$.

ПРИМЕР ПРОГРАММЫ НА РАЗРАБОТАННОМ ЯЗЫКЕ



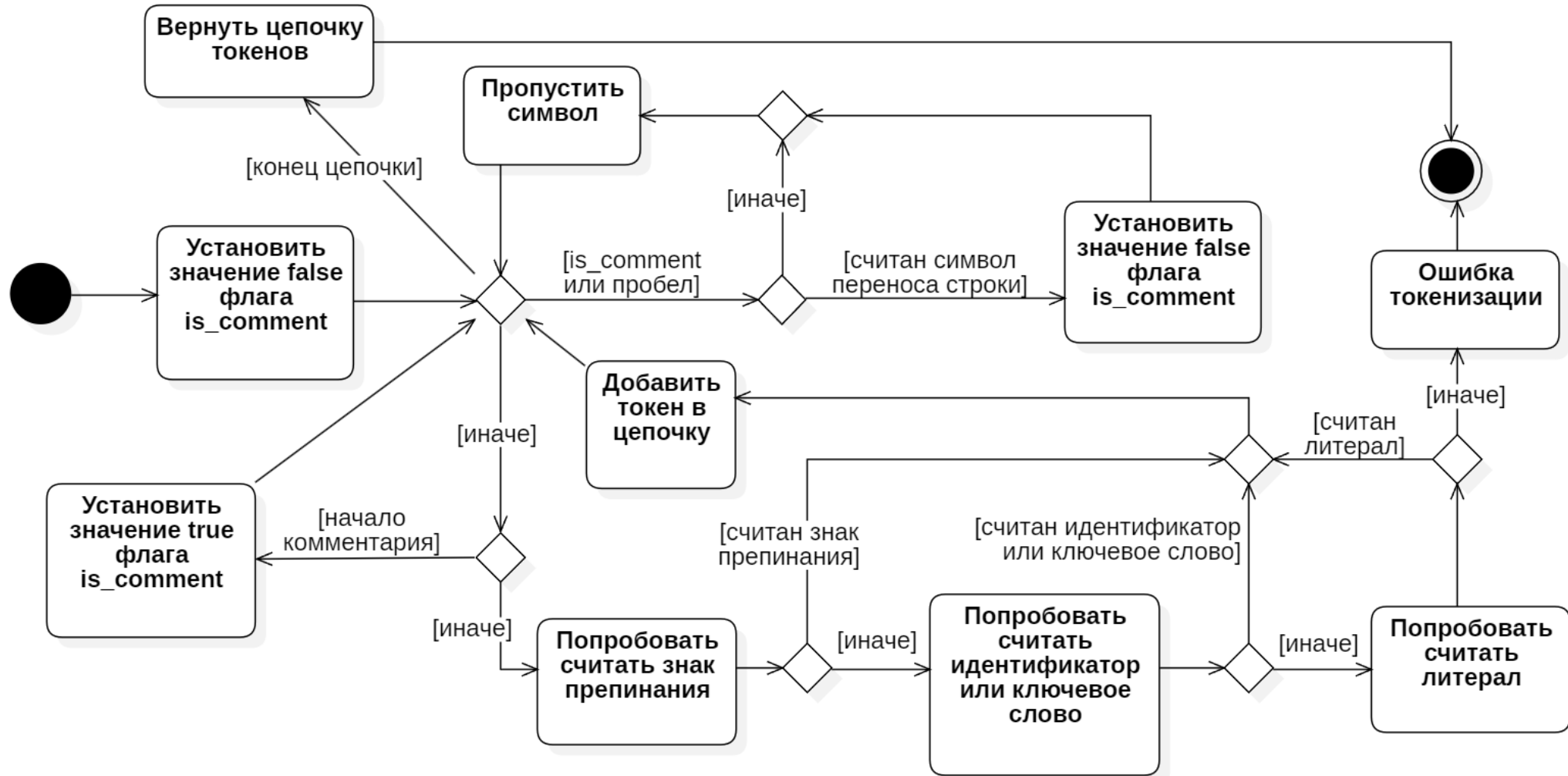
```
A: dec[0..9] = '0'..'9'.
Q: mul[0..1].

# Начальное состояние
start:
  dec{ _ } -> same, R, same;
  null -> same, L, mul[0].

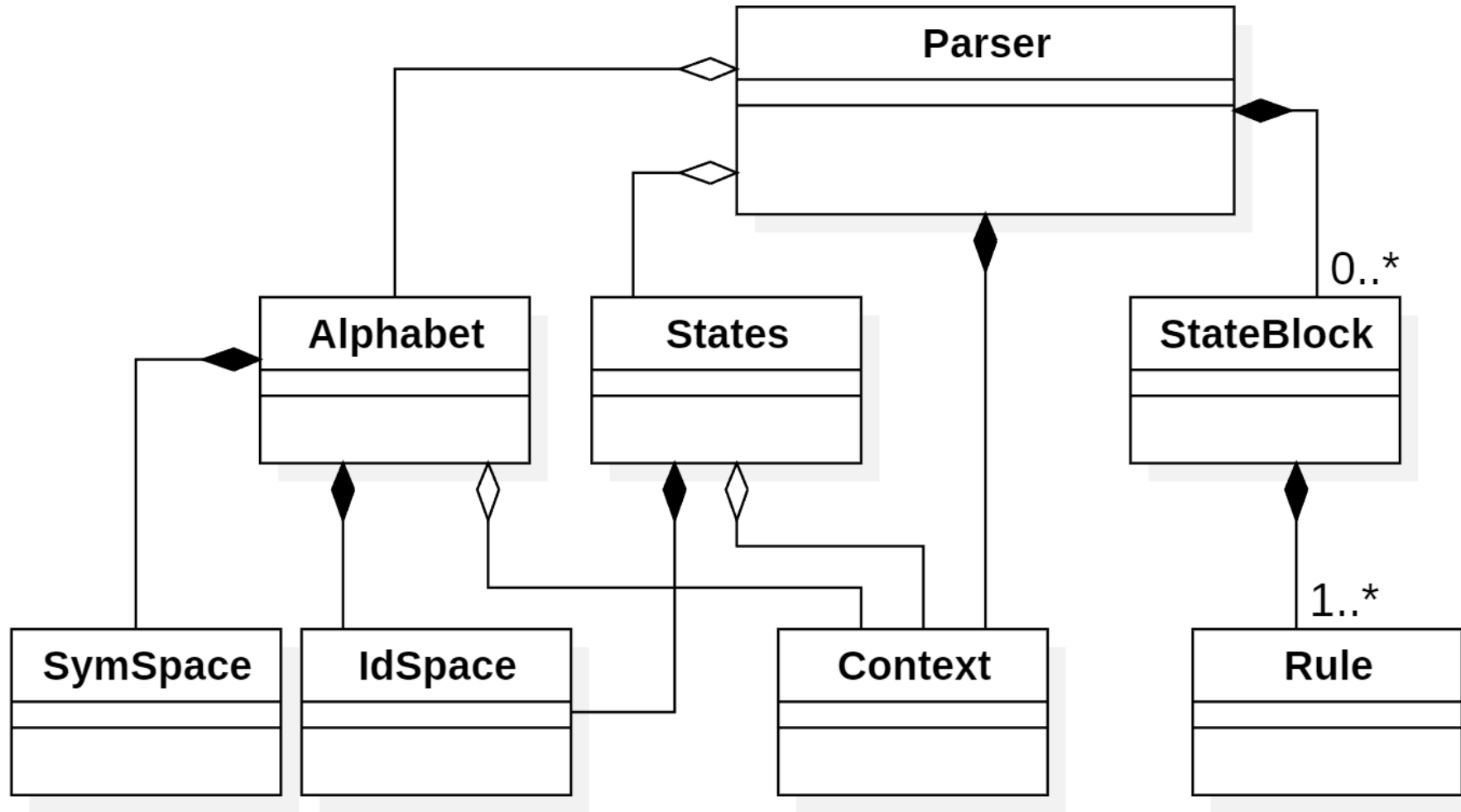
# Состояния mul[0] и mul[1]
mul{c}:
  dec{n} -> dec[(n * 2 + c) % 10], L, mul[n / 5].

mul[0]: null -> same, R, end.
mul[1]: null -> dec[1], N, end.
```


ДИАГРАММА ДЕЯТЕЛЬНОСТИ ТОКЕНИЗАТОРА



УПРОЩЁННАЯ ДИАГРАММА КЛАССОВ ПАРСЕРА



ПОЛНАЯ ДИАГРАММА КЛАССОВ ПАРСЕРА

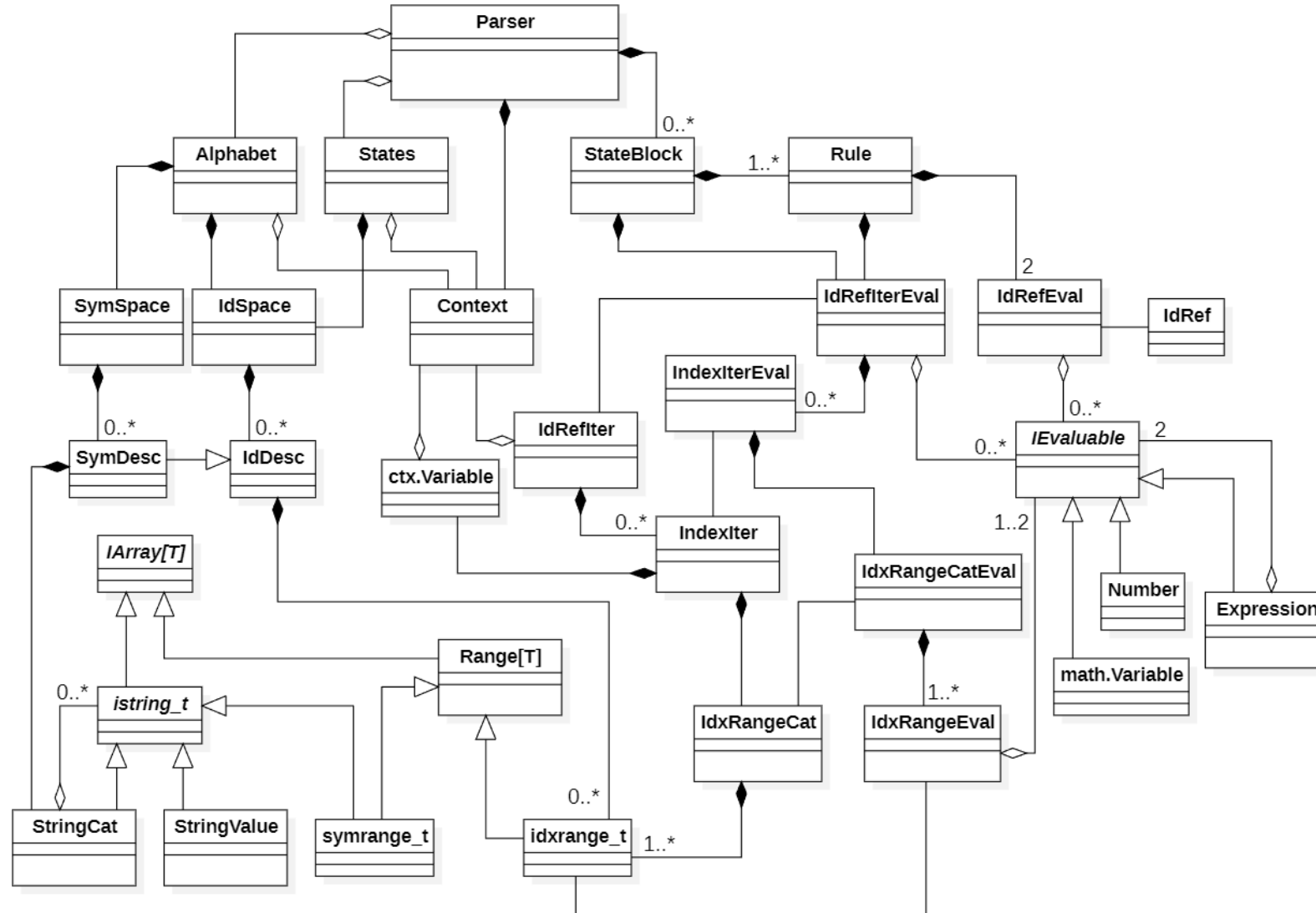
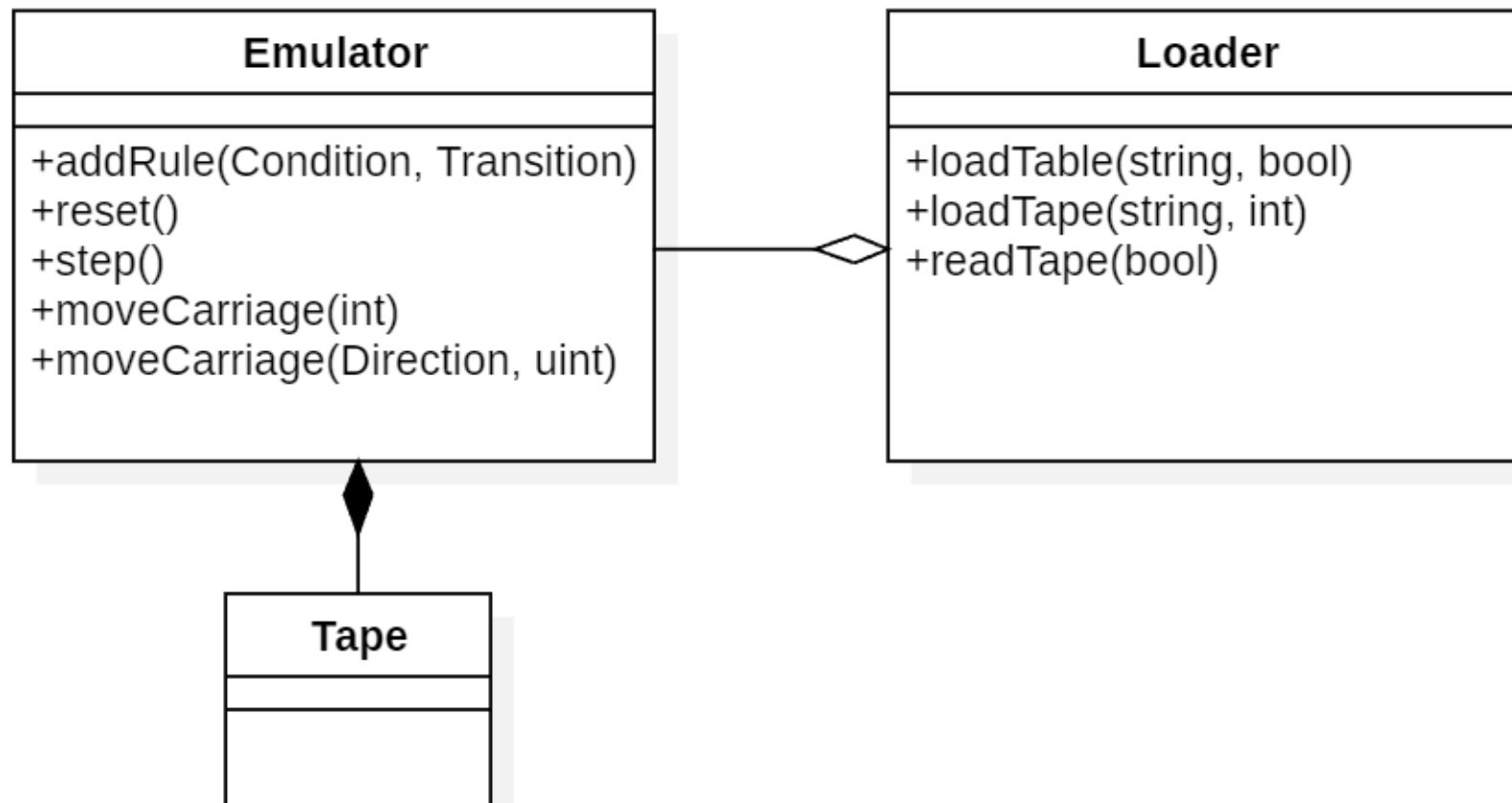


ДИАГРАММА КЛАССОВ ЭМУЛЯТОРА

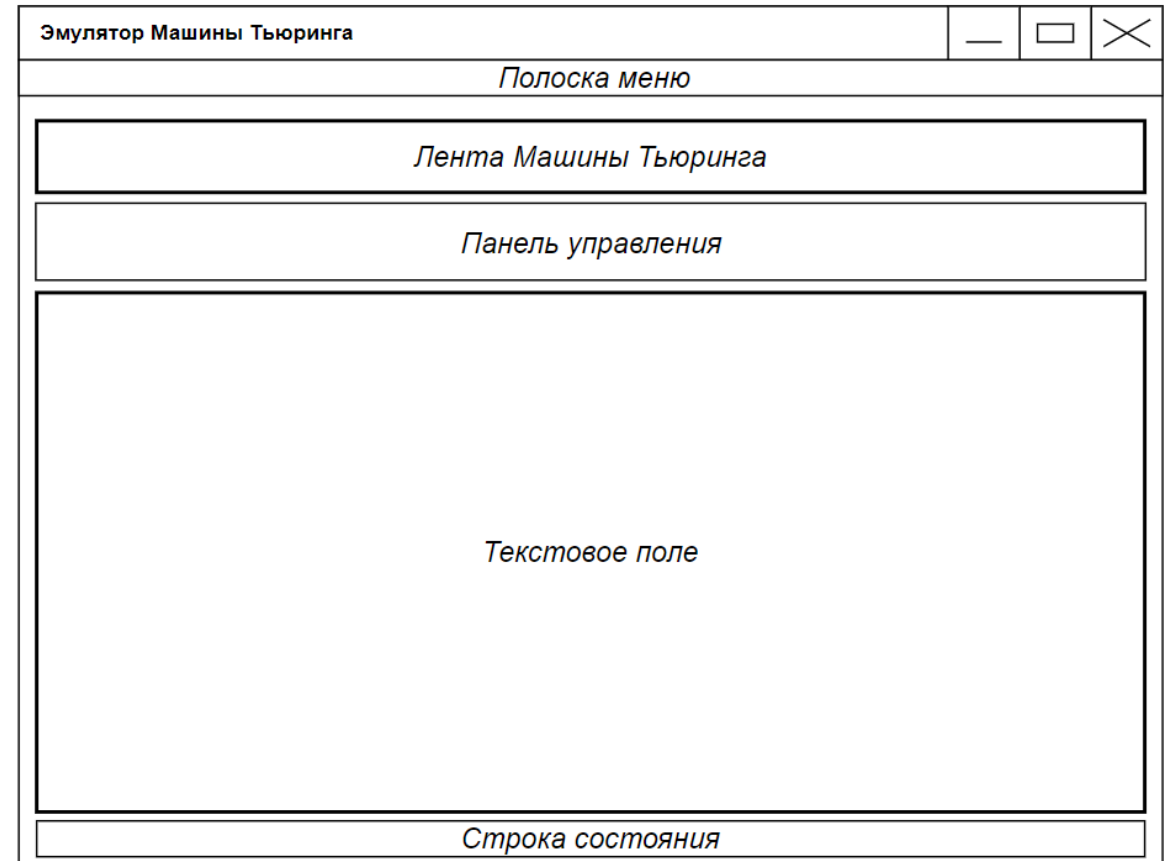


ПРОЕКТИРОВАНИЕ ИНТЕРФЕЙСА ЭМУЛЯТОРА



Программа-эмулятор должна:

- позволять редактировать файлы исходного кода;
- позволять загружать и выгружать данные в ленте в файл;
- позволять менять данные в ленте через диалоговое окно;
- отображать актуальное состояние ленты;
- позволять перемещать каретку вручную;
- отображать текущее состояние Машины Тьюринга;
- позволять запускать выполнение программы с заданной скоростью, с возможностью приостановить работу;
- позволять ручное пошаговое выполнение программы;
- позволять сбрасывать состояние машины и ленты.



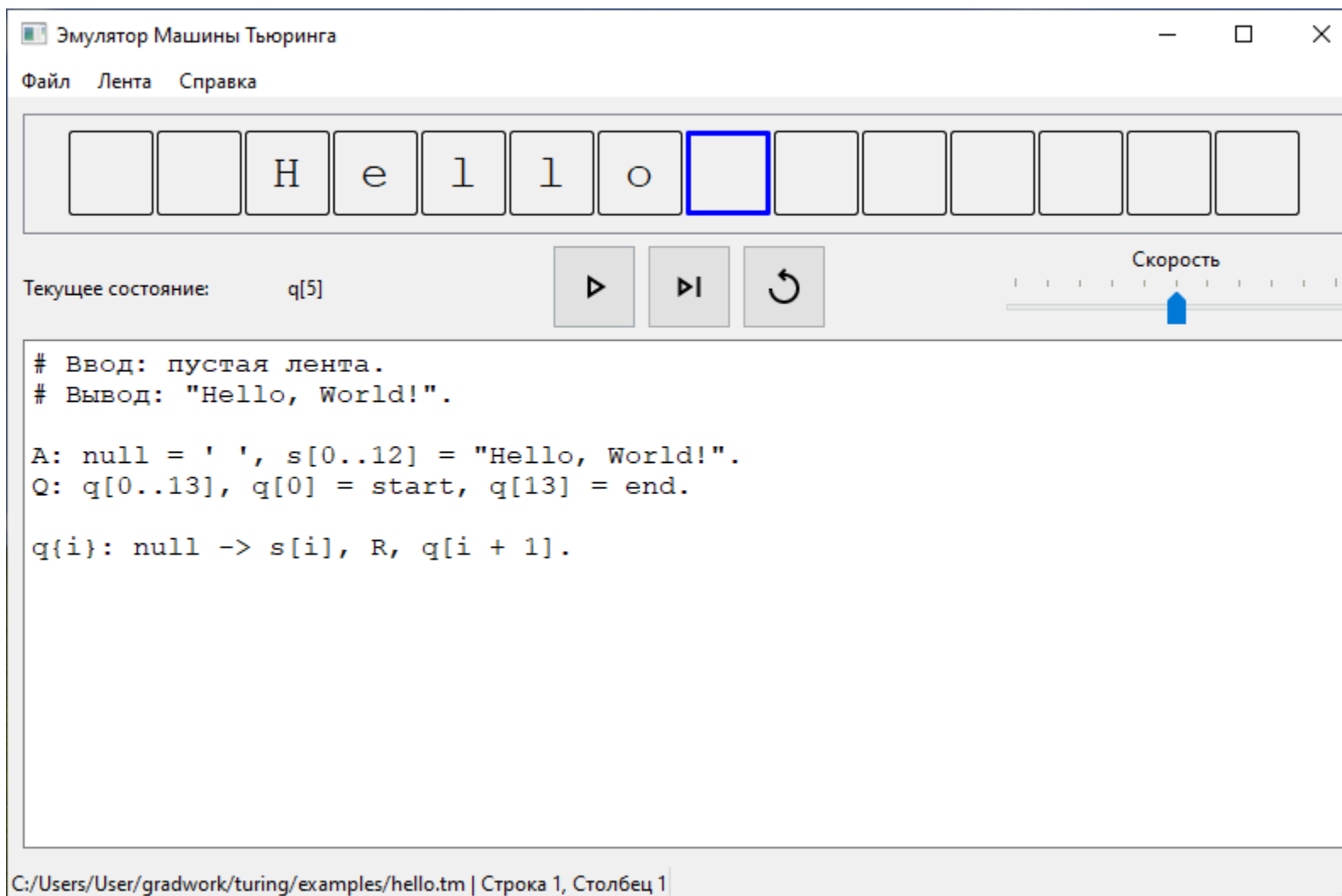
Макет графического интерфейса эмулятора

СРЕДСТВА РЕАЛИЗАЦИИ



- Язык программирования C++ (стандарт языка – C++20).
- Фреймворк QT 6.6.2.
- Модуль QtWidgets для реализации графического интерфейса.
- Среда разработки QtCreator 12.0.
- Собрание библиотек Boost 1.85.0.
- Система автоматизации сборки CMake 3.28.3.

ГРАФИЧЕСКИЙ ИНТЕРФЕЙС ЭМУЛЯТОРА



ТЕСТИРОВАНИЕ ИНТЕРПРЕТАТОРА



Проведенные тесты:

1. Объявление символов
2. Объявление состояний
3. Итерируемые ссылки и числовые последовательности
4. Арифметические операции
5. Строковые литералы и конкатенация строк
6. Алфавит не определен
7. Пропущен знак точки
8. Идентификатор не существует
9. Неправильно присвоено значение символу
10. Поведение не определено
11. Индекс выходит за пределы допустимого диапазона
12. Ошибка при присвоении значения символу null
13. Пропущена кавычка

ТЕСТИРОВАНИЕ ЭМУЛЯТОРА



Проведенные тесты:

1. Загрузка программы из файла
2. Загрузка данных в ленту из файла
3. Сохранение данных из ленты в файл
4. Отображение текущего состояния машины и ленты
5. Управление
6. Сброс
7. Изменение данных в ленте через диалоговое окно
8. Ручное перемещение каретки

РЕЗУЛЬТАТЫ



1. Выполнен анализ предметной области, изучены аналоги, разработаны требования к системе.
2. Спроектирован язык программирования для Машины Тьюринга, разработан интерпретатор.
3. Разработан эмулятор Машины Тьюринга, позволяющий выполнять программы, написанные на спроектированном языке, и визуализировать работу Машины Тьюринга.
4. Проведено тестирование интерпретатора, и функциональное тестирование эмулятора.