

Домашняя работа №3

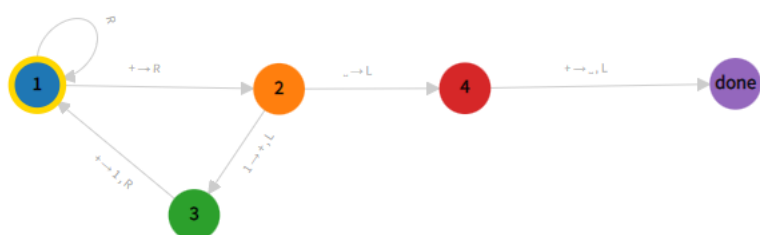
А-13а-19 Самсонова Мария

31 мая 2022 г.

1 Реализуйте машины Тьюринга, которые позволяют выполнять следующие операции:

1. Сложение двух унарных чисел

Граф решения МТ

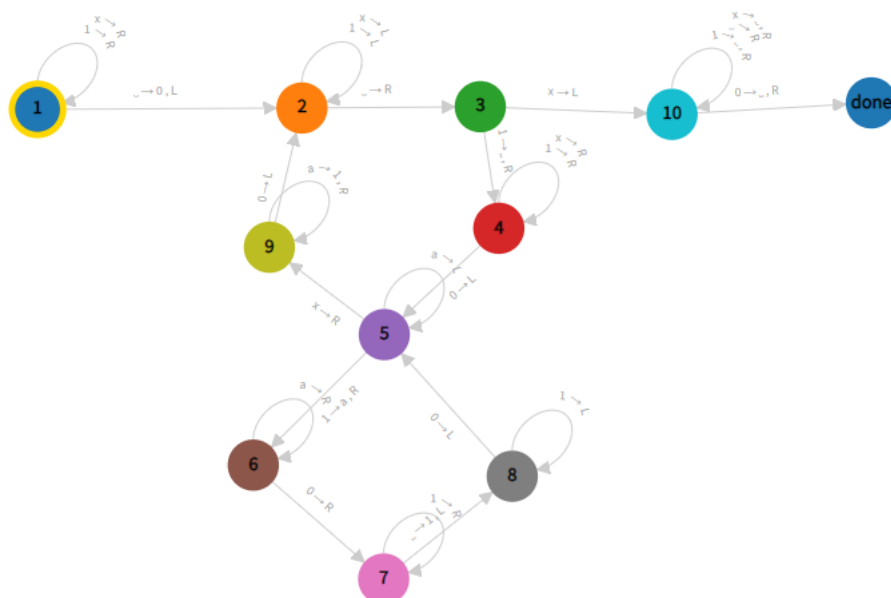


Проверка работоспособности алгоритма

№	входная строка	выходная строка											
1	<table><tr><td></td><td>1</td><td>1</td><td>+</td><td>1</td><td>1</td></tr></table>		1	1	+	1	1	<table><tr><td></td><td>1</td><td>1</td><td>1</td><td>1</td></tr></table>		1	1	1	1
	1	1	+	1	1								
	1	1	1	1									
2	<table><tr><td></td><td>+</td><td>1</td><td>1</td><td></td></tr></table>		+	1	1		<table><tr><td></td><td>1</td><td>1</td><td></td></tr></table>		1	1			
	+	1	1										
	1	1											
3	<table><tr><td></td><td>1</td><td>1</td><td>+</td><td></td></tr></table>		1	1	+		<table><tr><td></td><td>1</td><td>1</td><td></td></tr></table>		1	1			
	1	1	+										
	1	1											

2. Умножение унарных чисел

Граф решения МТ

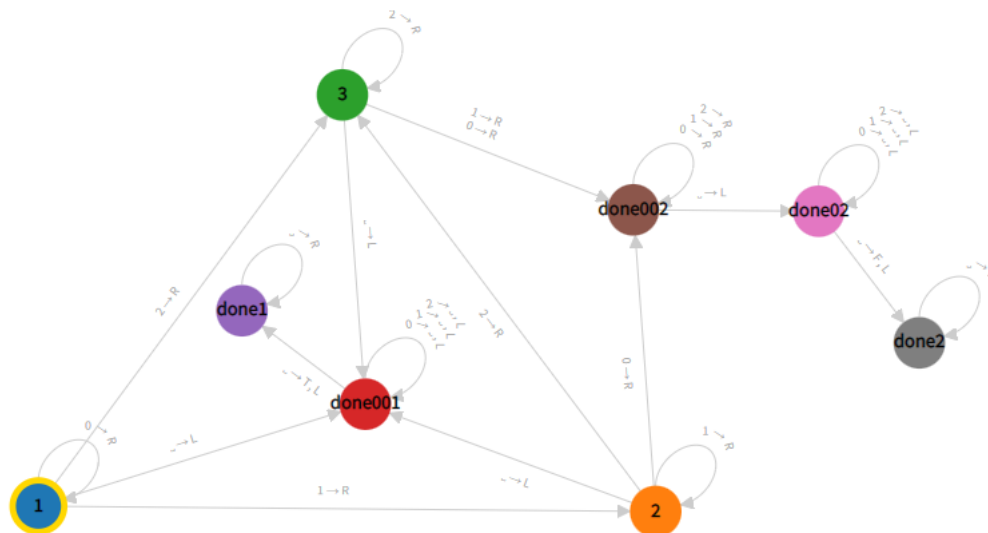


Проверка работоспособности алгоритма

№	входная строка	выходная строка																					
1	<table><tr><td></td><td></td><td></td><td></td><td>1</td><td>1</td><td>x</td><td>1</td><td>1</td><td>1</td><td></td></tr></table>					1	1	x	1	1	1		<table><tr><td></td><td></td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td></td><td></td></tr></table>			1	1	1	1	1	1		
				1	1	x	1	1	1														
		1	1	1	1	1	1																
2	<table><tr><td></td><td></td><td>x</td><td>1</td><td>1</td><td>1</td><td></td><td></td></tr></table>			x	1	1	1			<table><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>													
		x	1	1	1																		
3	<table><tr><td></td><td></td><td>1</td><td>1</td><td>1</td><td>x</td><td></td><td></td></tr></table>			1	1	1	x			<table><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>													
		1	1	1	x																		

3. Принадлежность к языку $L = \{0^n 1^n 2^n\}, n \geq 0$

Граф решения МТ

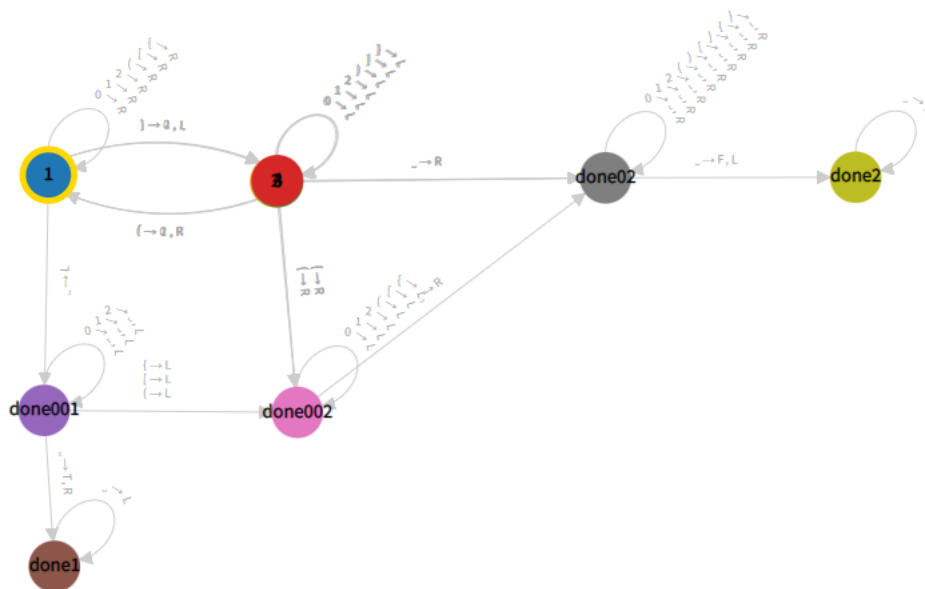


Проверка работоспособности алгоритма

№	входная строка	выходная строка														
1	<table><tr><td></td><td>0</td><td>0</td><td>1</td><td>1</td><td>2</td><td></td></tr></table>		0	0	1	1	2		<table><tr><td></td><td></td><td>T</td><td></td><td></td><td></td><td></td></tr></table>			T				
	0	0	1	1	2											
		T														
2	<table><tr><td></td><td>0</td><td>0</td><td>2</td><td></td><td></td><td></td></tr></table>		0	0	2				<table><tr><td></td><td></td><td>T</td><td></td><td></td><td></td><td></td></tr></table>			T				
	0	0	2													
		T														
3	<table><tr><td></td><td>1</td><td>1</td><td></td><td></td><td></td><td></td></tr></table>		1	1					<table><tr><td></td><td></td><td>T</td><td></td><td></td><td></td><td></td></tr></table>			T				
	1	1														
		T														
4	<table><tr><td></td><td>1</td><td>2</td><td>1</td><td></td><td></td><td></td></tr></table>		1	2	1				<table><tr><td></td><td></td><td>F</td><td></td><td></td><td></td><td></td></tr></table>			F				
	1	2	1													
		F														
5	<table><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>								<table><tr><td></td><td></td><td>T</td><td></td><td></td><td></td><td></td></tr></table>			T				
		T														

4. Проверка соблюдения правильности скобок в строке (минимум 3 вида скобок)

Граф решения МТ

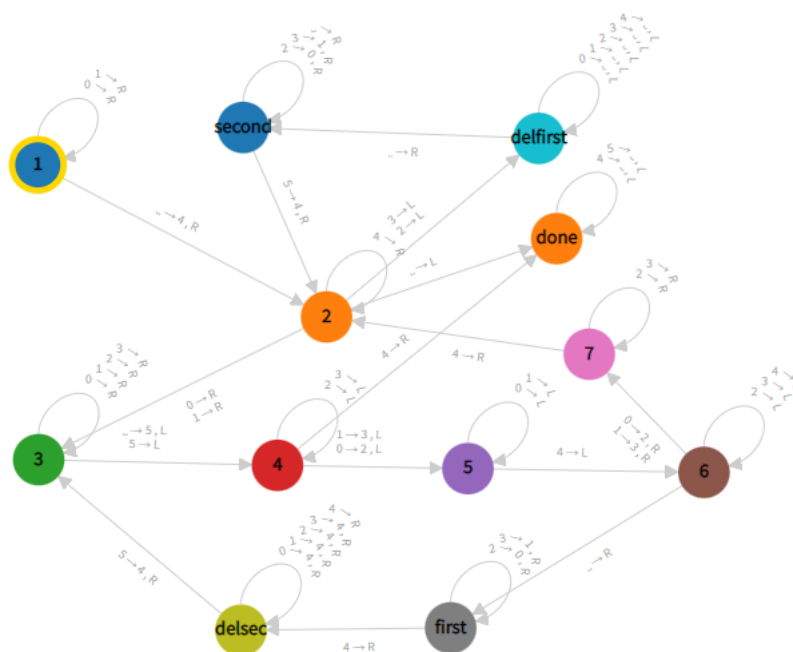


Проверка работоспособности алгоритма

№	входная строка	выходная строка
1	[[{ [] () }]]	[T]
2	[([)]]	[F]
3	[[()]]	[F]
4	[[() }]]	[F]

5. Поиск минимального по длине слова в строке (слова состоят из символов 1 и 0 и разделены пробелом)

Граф решения МТ



Проверка работоспособности алгоритма

№	входная строка	выходная строка																				
1	<table><tr><td></td><td>1</td><td>1</td><td>0</td><td>1</td><td></td><td>1</td><td>0</td><td>1</td><td></td></tr></table>		1	1	0	1		1	0	1		<table><tr><td></td><td></td><td>1</td><td>0</td><td>1</td><td></td><td></td><td></td><td></td><td></td></tr></table>			1	0	1					
	1	1	0	1		1	0	1														
		1	0	1																		
2	<table><tr><td></td><td></td><td>1</td><td>1</td><td></td><td></td><td>1</td><td>0</td><td>1</td><td></td></tr></table>			1	1			1	0	1		<table><tr><td></td><td></td><td>1</td><td>1</td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>			1	1						
		1	1			1	0	1														
		1	1																			
3	<table><tr><td>1</td><td>1</td><td></td><td></td><td>1</td><td></td><td>1</td><td>0</td><td></td><td>0</td></tr></table>	1	1			1		1	0		0	<table><tr><td></td><td></td><td></td><td>0</td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>				0						
1	1			1		1	0		0													
			0																			
4	<table><tr><td></td><td></td><td></td><td>1</td><td>1</td><td></td><td></td><td></td><td></td><td></td></tr></table>				1	1						<table><tr><td></td><td></td><td>1</td><td>1</td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>			1	1						
			1	1																		
		1	1																			

2 Квантовые вычисления

Дано N кубитов ($1 \leq N \leq 8$) в нулевом состоянии $0 \dots 0$. Также дана некоторая последовательность битов, которое задаёт ненулевое базисное состояние размера N . Задача получить суперпозицию нулевого состояния и заданного.

$$S = \frac{1}{\sqrt{2}}(0 \dots 0 + \psi)$$

То есть требуется реализовать операцию, которая принимает на вход:

1. Массив кубитов q_s
2. Массив битов $bits$ описывающих некоторое состояние ψ . Это массив имеет тот же самый размер, что и q_s . Первый элемент этого массива равен 1.

```
namespace Quantum.HW3
```

```
{
```

```
    open Microsoft.Quantum.Canon;
    open Microsoft.Quantum.Intrinsic;
    open Microsoft.Quantum.Diagnostics;
    open Microsoft.Quantum.Measurement;
```

```
    @EntryPoint()
```

```
    operation SolveForTwoQubits() : Result[]
```

```
    {
```

```
        using (qubits = Qubit[1])
```

```
        {
```

```
            DumpMachine();
```

```
            Solve(qubits);
```

```
            DumpMachine();
```

```
            let results = MultiM(qubits);
```

```
            ResetAll(qubits);
```

```
            return results;
```

```
        }
```

```
    }
```

```
    operation Solve (qs : Qubit[]) : ()
```

```
    {
```

```
        body
```

```
        {
```

```
            for (q in qs)
```

```
            {
```

```

        if (Zero != M(q))
        {
            X(q);
        }
    }
    H(qs[0]);

    for (i in 1..Length(qs) - 1)
    {
        use q = Qubit();
        if (Zero != M(q))
        {
            CNOT(qs[0], qs[i]);
        }
    }
}
}

```