

Теоретические модели вычислений

ДЗ №3: Машины Тьюринга и квантовые вычисления

1 мая 2022 года

1 Введение

Для того, чтобы работа была принята, требуется соблюдать ряд условий:

1. Текст решения домашней работы должен быть подготовлен в одном из следующих форматов: [L^AT_EX](#), [Markdown](#), [AsciiDOC](#)
2. Дополнительно должен быть приложен PDF файл
3. Для набора рекомендуется использовать сервисы типа [Overleaf](#)
4. Работу можно выполнить на английском языке
5. Иллюстрации, должны быть описаны на декларативном языке [Graphviz](#) или [Mermaid](#). В сам документ можно вставить отрендеренную картинку.
6. За каждое задание можно получить определённое количество баллов, указанное рядом с заданием
7. Мягкий дедлайн – 20 дней (12:00 20 мая 2022 года), после него –30% к оценке
8. Для зачёта надо набрать 60% от максимальной оценки
9. Максимальное количество баллов: 10.
10. Работа должна быть выложена на [GitHub](#), в репозитории, созданном к ней

2 Машины Тьюринга

Работу требуется выполнять в системе [turingmachine.io](#).

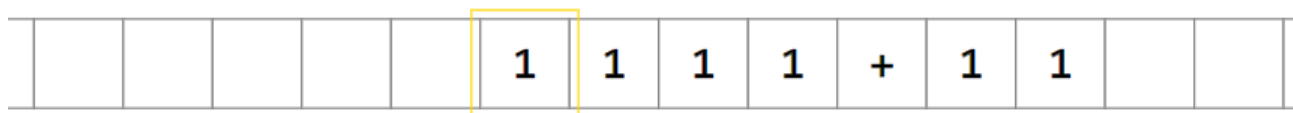
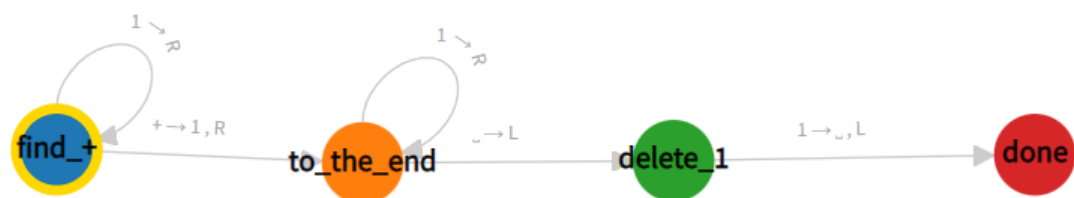
Для сдачи заданий 1-2 требуется прикрепить файлы YAML с исходным кодом проекта. Каждый файл должен иметь наименование задание_пункт.yml, к примеру 1_1.yml для первой задачи первого задания.

2.1 Операции с числами

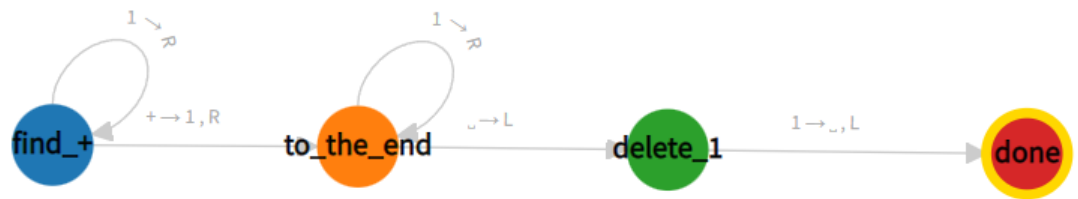
Реализуйте машины Тьюринга, которые позволяют выполнять следующие операции:

1. Сложение двух унарных чисел (1 балла)

Начало

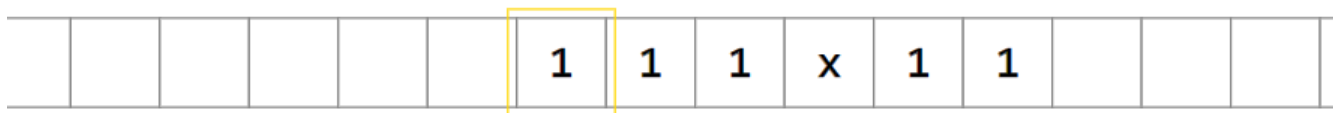
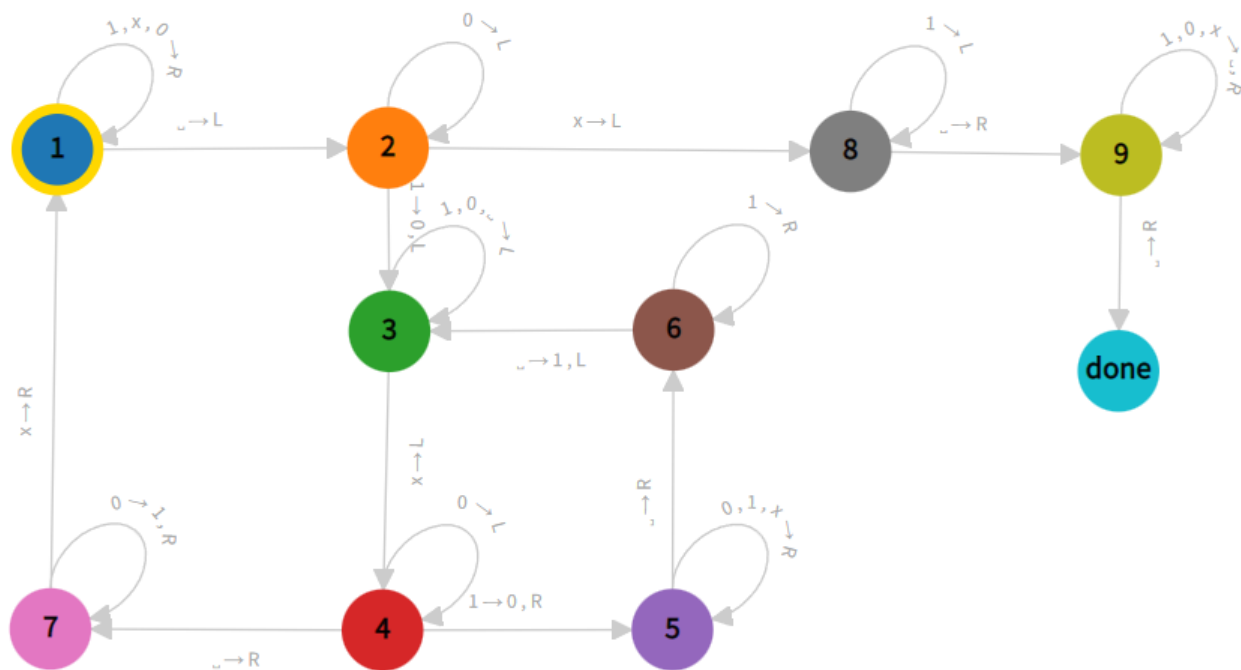


Конец

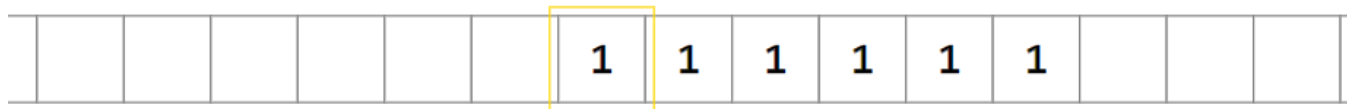
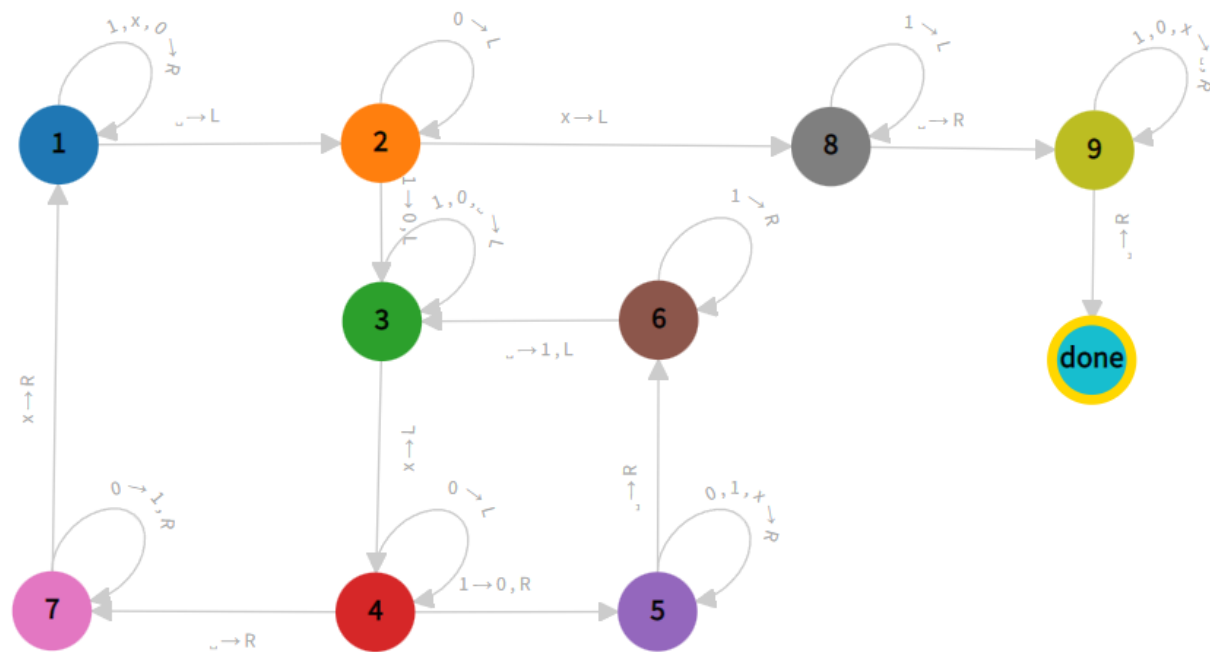


2. Умножение унарных чисел (1 балл)

Начало



Конец

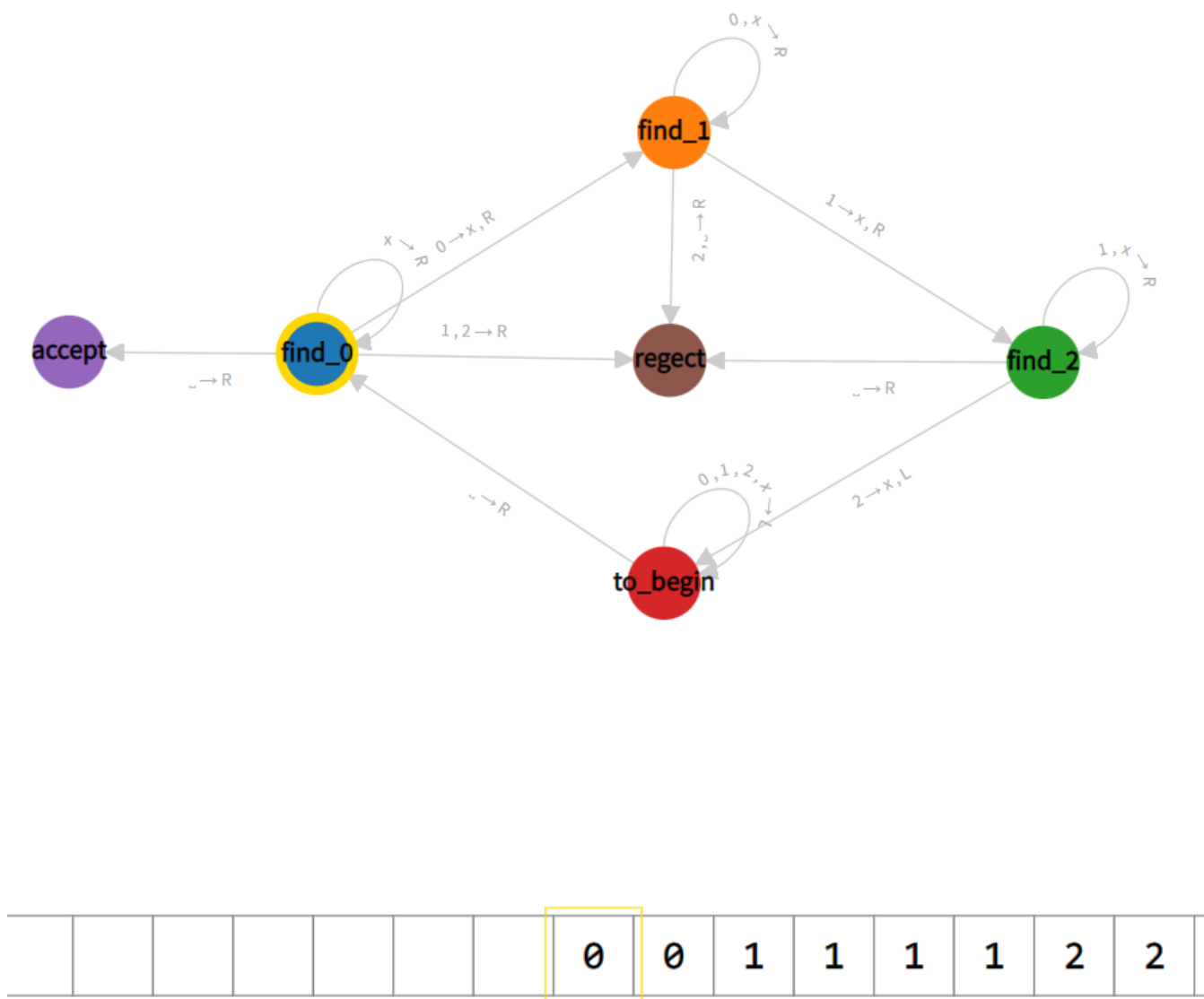


2.2 Операции с языками и символами

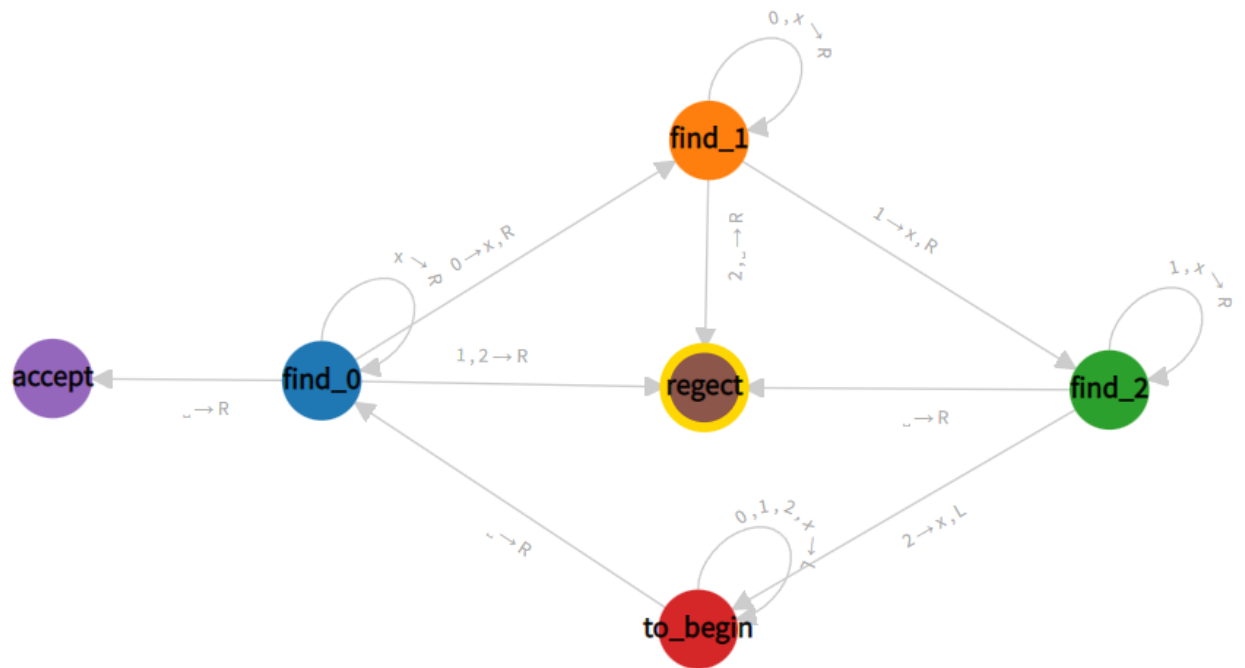
Реализуйте машины Тьюринга, которые позволяют выполнять следующие операции:

1. Принадлежность к языку $L = \{0^n 1^n 2^n\}, n \geq 0$ (0.5 балла)

Начало

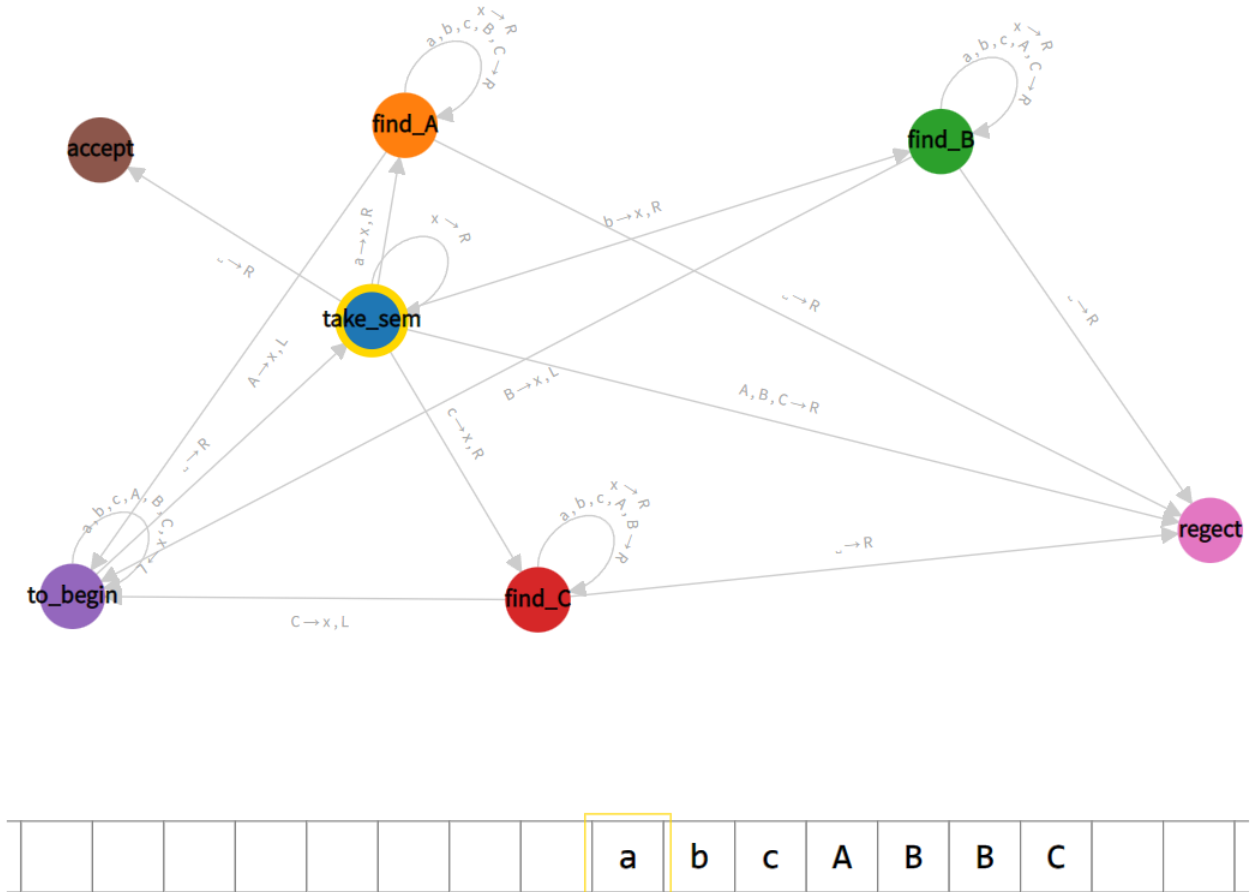


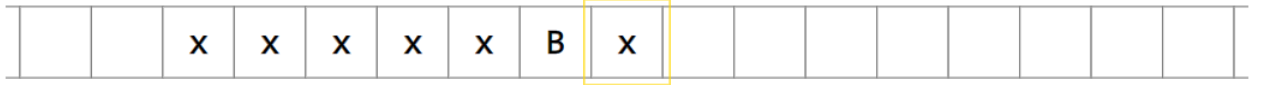
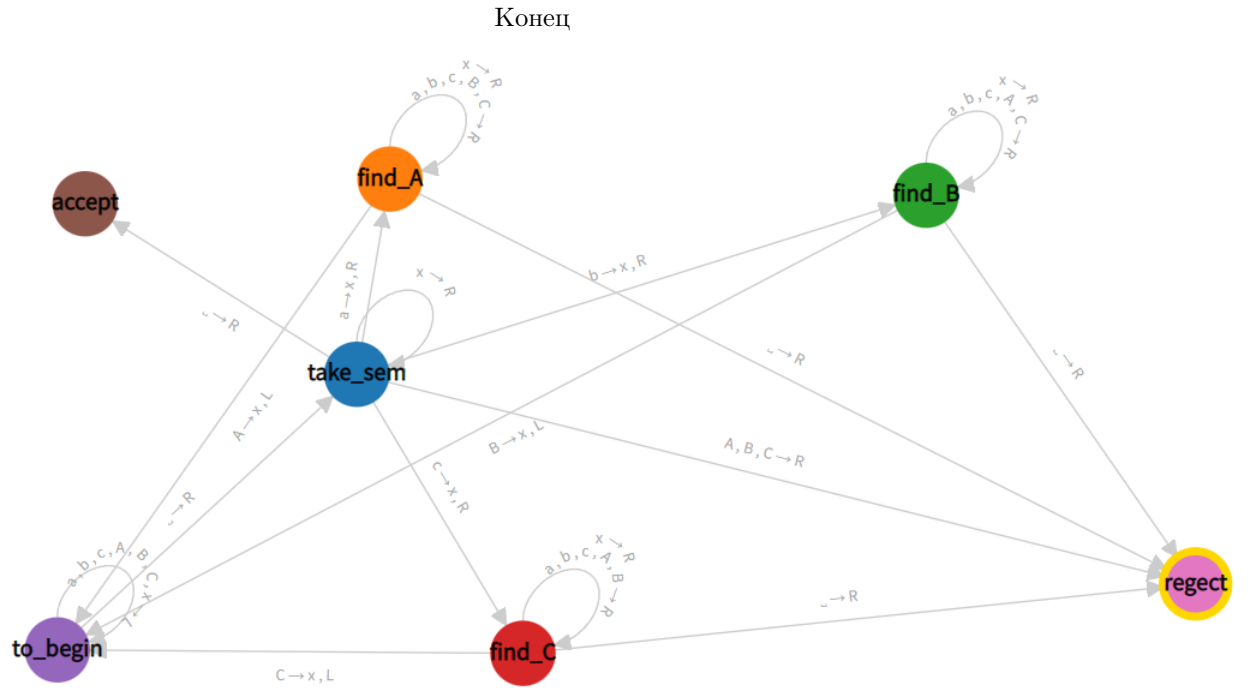
Конец



2. Проверка соблюдения правильности скобок в строке (минимум 3 вида скобок) (0.5 балла)
 Была какая-то проблема с записью скобок, по тому я решил сделать с буквами, суть задания не поменялась

Начало





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

3 Квантовые вычисления

Для выполнения заданий по квантовым вычислениям требуется QDK. Его можно скачать здесь: <https://docs.microsoft.com/en-us/azure/quantum/install-overview-qdk>.

Но можно использовать любой пакет, типа <https://qiskit.org/>.

В качестве решения задачи надо предоставить схему алгоритма для частного случая при фиксированном количестве кубитов и фиксированных состояниях.

3.1 Генерация суперпозиций 1 (1 балл)

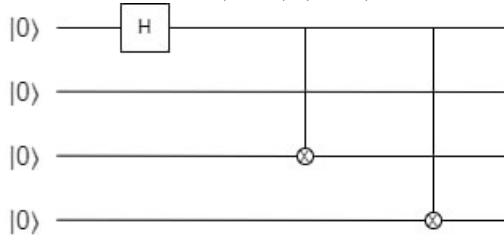
Дано N кубитов ($1 \leq N \leq 8$) в нулевом состоянии $|0 \dots 0\rangle$. Также дана некоторая последовательность битов, которое задаёт ненулевое базисное состояние размера N . Задача получить суперпозицию нулевого состояния и заданного.

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

То есть требуется реализовать операцию, которая принимает на вход:

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

Схема для случая $|0000\rangle; |1011\rangle$:



Код:

```
namespace Solution {
    open Microsoft.Quantum.Primitive;
    open Microsoft.Quantum.Canon;
    operation Solve (qs : Qubit[], bits : Bool[]) : ()
    {
        body
        {
            H(qs[0]);
            for (i in 1..Length(qs) - 1) {
                if (bits[i]) {
                    CNOT(qs[0], qs[i]);
                }
            }
        }
    }
}
```

3.2 Различение состояний 1 (1 балл)

Дано N кубитов ($1 \leq N \leq 8$), которые могут быть в одном из двух состояний:

$$|GHZ\rangle = \frac{1}{\sqrt{2}}(|0\dots 0\rangle + |1\dots 1\rangle)$$

$$|W\rangle = \frac{1}{\sqrt{N}}(|10\dots 00\rangle + |01\dots 00\rangle + \dots + |00\dots 01\rangle)$$

Требуется выполнить необходимые преобразования, чтобы точно различить эти два состояния. Возвращать 0, если первое состояние и 1, если второе.

Заготовка для кода:

```
namespace Solution {
    open Microsoft.Quantum.Primitive;
    open Microsoft.Quantum.Canon;
    operation Solve (qs : Qubit[]) : Int
    {
        body
```

```

    {
        mutable countOnes = 0;
        for (i in 0..Length(qs) - 1) {
            if (M(qs[i]) = One){
                set countOnes = countOnes + 1;
            }
        }
        if (countOnes = Length(qs) or countOnes = 0) {
            return 0;
        }
        return 1;
    }
}

```

3.3 Различение состояний 2 (2 балла)

Дано 2 кубита, которые могут быть в одном из двух состояний:

$$|S_0\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$$

$$|S_1\rangle = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle)$$

$$|S_2\rangle = \frac{1}{2}(|00\rangle + |01\rangle - |10\rangle - |11\rangle)$$

$$|S_3\rangle = \frac{1}{2}(|00\rangle - |01\rangle - |10\rangle + |11\rangle)$$

Требуется выполнить необходимые преобразования, чтобы точно различить эти четыре состояния. Возвращать требуется индекс состояния (от 0 до 3).

Заготовка для кода:

```

namespace Solution {
    open Microsoft.Quantum.Primitive;
    open Microsoft.Quantum.Canon;
    operation Solve (qs : Qubit[]) : Int
    {
        body
        {
            return
        }
    }
}

```

3.4 Написание оракула 1 (2 балла)

Требуется реализовать квантовый оракул на N кубитах ($1 \leq N \leq 8$), который реализует следующую функцию: $f(\mathbf{x}) = (\mathbf{bx}) \bmod 2$, где $\mathbf{b} \in \{0, 1\}^N$ вектор битов и \mathbf{x} вектор кубитов. Выход функции записать в кубит \mathbf{y} . Количество кубитов N ($1 \leq N \leq 8$).

Заготовка для кода:

```
namespace Solution {  
    open Microsoft.Quantum.Primitive;  
    open Microsoft.Quantum.Canon;  
    operation Solve (x : Qubit[], y : Qubit, b : Int[]) : ()  
    {  
        body  
        {  
        }  
    }  
}
```