

"Теоретические модели вычислений ДЗ№3"
Бородин Сергей Владимирович А-13а-19

Теоретические модели вычислений

ДЗ №3: Машины Тьюринга и квантовые вычисления

1 мая 2022 года

1 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.

1.1 Операции с числами

Реализуйте машины Тьюринга, которые позволяют выполнять следующие операции:

1. Сложение двух унарных чисел (1 балла)

(Алгоритм: Машина Тьюринга принимает строку состоящую из двух унарных чисел разделенных +. Машина будет переносить посимвольно первое число в конец второго числа, а в конце когда в начале будет + затрет его и закончит работу. Исходный код лежит в файле 1-1. yaml)

2. Умножение унарных чисел (1 балл)

(Алгоритм: Машина Тьюринга принимает строку состоящую из двух унарных чисел разделенных *. Машина будет затирать один символ первого числа, а затем копировать все единицы второго числа в после второго числа, этот процесс будет повторяться пока первое число не затрется. После Машина затрет второе число и знак * и завершит работу. Исходный код лежит в файле 1-2. yaml)

1.2 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)

(Алгоритм: Машина Тьюринга принимает строку состоящую из 0 1 2. Машина будет искать 0, превращать его в *, возвращаться в начало после искать 1 и возвращаться в начало, после искать 2 и возвращаться в начало. В случае, если при поиске 0 будет найден ' ', то проверяется, что все предыдущие элементы строки *, и в этом случае выводится 'Т'. В случае, если ' ' найден в случае поиска 1 или 2, то выводится 'F', так же оно выводится, если

при проверке строки на '*', найдет какой-то другой элемент, то в конце тоже выводится 'F'. После вывода 'T' или 'F', машина завершает работу. Исходный код лежит в файле 2-1. yaml)

2. Проверка соблюдения правильности скобок в строке (минимум 3 вида скобок) (0.5 балла)
(Алгоритм: Машина Тьюринга принимает строку состоящую из ([]). Машина будет искать закрывающую скобку и менять ее на символ, потом идти налево и искать соответствующую открывающую ее тоже менять на символ, при этом, если будет находить ошибочно открывающую, то ставить N и затирает все кроме нее. Процесс будет повторяться, пока уйдут все скобки, либо не выявится ошибка. Исходный код лежит в файле 2-2-1. yaml)
3. Поиск минимального по длине слова в строке (слова состоят из символов 1 и 0 и разделены пробелом) (1 балл)

(Алгоритм: Машина Тьюринга принимает строку состоящую из 1 0 ' '. Машина будет искать Машина будет идти до конца строки и ставит флажок, далее возвращать назад и идти по каждому слову и менять на первую букву слова на символ, потом процесс повторится для вторых букв и так далее, в итоге, когда вместо буквы будет найден ' ', слово будет "помечено а остальные слова затерты, полученное слово приведетс в нормальный вид. Исходный код лежит в файле 2-3. yaml)

2 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.

2.1 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.

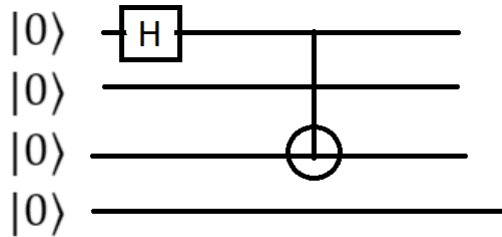
```
operation Solve (qs: Qubit [], bits: Bool []) : ()
{
  body
  {
```

```

H(qs[0]);
for (i in 1..Length(qs) - 1)
{
    if (bits[i])
    {
        CNOT(qs[0], qs[i]);
    }
}
}

```

Для входных данных $|0\ 0\ 0\ 0\rangle, |1\ 0\ 1\ 0\rangle$ схема алгоритма будет выглядеть следующим образом:



2.2 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.

```

operation Solve (qs: Qubit[]) : Int
{
    body
    {
        mutable countOnes = 0;
        for (i in 0..Length(qs) - 1)
        {
            if (M(qs[i]) == One)
            {
                set counter = counter + 1;
            }
        }
        if (counter == 1)
        {
            return 1;
        }
    }
}

```

```

    }
    return 0;
  }
}

```

2.3 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).

```

operation Solve (qs: Qubit[]) : Int
{
  body
  {
    H(qs[0]);
    H(qs[1]);
    if (M(qs[0]) == Zero)
    {
      if (M(qs[1]) == Zero)
      {
        return 0;
      }
      else
      {
        return 1;
      }
    }
    else
    {
      if (M(qs[1]) == Zero)
      {
        return 2;
      }
      else
      {
        return 3;
      }
    }
  }
}

```

2.4 Написание оракула 1 (2 балла)

Требуется реализовать квантовый оракул на N кубитах ($1 \leq N \leq 8$), который реализует следующую функцию: $f(\mathbf{x}) = (\mathbf{b}\mathbf{x}) \bmod 2$, 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  
        {  
        }  
    }  
}
```