**ЛАБОРАТОРНАЯ РАБОТА № 1**

***«Определение машины Тьюринга. Применение машины Тьюринга к словам. Конструирование машин Тьюринга»***

**Цель работы:** получить практические навыки в реализации алгоритмов с использованием машин Тьюринга.

**Теоретический материал**

**Символьные конструкции**

**Алфавитом** будем называть любое конечное множество попарно различных знаков, называемых **буквами** (символами) этого алфавита. Алфавит будем обозначать заглавными буквами, например:



Символом *λ* будем обозначать пустой символ.

**Словом**в данном алфавите называется любая конечная (в том числе и пустая) последовательность букв этого алфавита. Слова будем обозначать малыми греческими буквами.

Например: *α* = алгоритм – слово в алфавите *А*; *β* = 1010100 – слово в алфавите *В*;  – слово в алфавите *С*.

Пустое слово будем обозначать *Λ*.

**Длина** слова *α* (обозначается *|α|*) – это количество букв в слове.

Определим некоторые отношения и операции над словами.

**Равенство слов**в алфавите *А* определяется индуктивно:

а) пустые слова равны

б) если слово α равно слову β, то α *b* =β*b* , где *b* –буква в алфавите А.

Если слово *α* является частью слова *β*, то говорят, что имеет место **вхождение** слова *α* в слово *β* (слово *α* называется подсловом слова *β*). Это можно записать следующим образом: , где  – слова в алфавите *А.*

Слово *α* называется **началом слова** *β*, если ; **концом** слова *β*, если . Слово длины *n*, составленное из буквы *а*, повторенной *n* раз, будем обозначать , например *xyxxxyyyy = .*

Операция (и результат) приписывания слов *α* и *β* друг к другу называется **конкатенацией** (обозначается *α||β*). Например, если .

**Определение машины Тьюринга (МТ)**

Под машиной Тьюринга понимается некоторая гипотетическая (абстрактная) машина, состоящая из следующих частей:

1. бесконечной в обе стороны ленты, разбитой на ячейки, в каждой из ячеек может быть записан только один символ из алфавита , а также пустой символ *λ*;
2. рабочей головки или управляющего устройства (УУ), которое в каждый момент времени может находиться в одном из состояний множества . В каждом из состояний головка размещается напротив ячейки и может считывать (обозревать) или записывать в нее букву из алфавита *А*.

*λ*



......



......



*λ*

......

......



*λ*

Машина Тьюринга

Функционирование МТ состоит из последовательности элементарных шагов (тактов). На каждом шаге выполняются следующие действия:

1. управляющее устройство считывает (обозревает) символ ;
2. в зависимости от своего состояния  и обозреваемого символа 

УУ вырабатывает символ  и записывает его в обозреваемую ячейку (возможно );

1. головка перемещается на одну ячейку вправо *(R)*, влево *(L)* или остается на месте *(E)*;
2. головка переходит в другое внутреннее состояние  (возможно ).

Состояние  называется начальным,  – заключительным. При переходе в заключительное состояние машина останавливается.

Полное состояние МТ называется *конфигурацией*. Это распределение букв по ячейкам ленты, состояние рабочей головки и обозреваемая ячейка. Конфигурация в такте *t* записывается в виде: , где  – подслово слева от обозреваемой ячейки,  – буква в обозреваемой ячейке, – подслово справа. Начальная конфигурация  и конечная  называются стандартными.

Для описания работы МТ существует 3 способа:

1) система команд вида 

2) функциональная таблица;

3) граф (диаграмма) переходов.

С помощью МТ можно описывать выполнение арифметических операций над числами. При этом числа представляются на ленте, как слова в алфавите, состоящем из цифр какой-нибудь системы счисления, и разделяющихся специальным знаком, не входящем данный алфавит, например, .

Наиболее употребительной является унарная система, состоящая из одного символа  – . Число *Х* в унарной системе счисления на ленте записывается словом , ( сокращенно ) в алфавите *А={*  *}*.

*Пример 1.* Операция сложения двух чисел в унарном коде.

Начальная конфигурация: . Конечная конфигурация: , т.е. сложение фактически сводится к приписыванию числа *b* к числу *a* . Для этого первый символ  стирается, а *\** заменяется на .

Система команд при  и .



Комментарий к системе команд

1.  – стирание первого символа .

Если в обозреваемой ячейке записан символ  и МТ находится в состоянии , тогда состояние изменяется на , обозреваемый символ заменяется на пустой, УУ сдвигается вправо.

2. – стирание символа , первый аргумент равняется 0.

Если в обозреваемой ячейке записан символ  и МТ в состоянии  (первый аргумент равняется 0), тогда состояние изменяется на , обозреваемый символ заменяется на пустой, УУ сдвигается вправо.

3*.*  – сдвиг вправо.

Если в обозреваемой ячейке записан символ, записан символ  и МТ находится в состоянии , тогда состояние и обозреваемый символ не изменяются, УУ сдвигается вправо.

4.  – стирание символа .

Если в обозреваемой ячейке записан символ  и МТ находится в состоянии , тогда состояние изменяется на , и обозреваемый символ заменяется на , УУ сдвигается влево (конец первого аргумента).

5.  – сдвиг влево.

Если в обозреваемой ячейке записан символ  и МТ находится в состоянии ,тогда состояние и обозреваемый символ не изменяются, УУ сдвигается влево.

6.  –

Если в обозреваемой ячейке записан символ  и МТ находится в состоянии , тогда состояние изменяется на , обозреваемый символ не изменяется, УУ сдвигается вправо (конец алгоритма, УУ расположено в начале рабочей зоны ).

Описание МТ в виде функциональной таблицы:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **|** | **\*** | **λ** |
|  |  |  | - |
|  |  |  | - |
|  |  | - |  |

Описание МТ в виде диаграммы переходов













Вычисление на МТ словарной функции  будем понимать следующим образом. Пусть в начальной конфигурации на ленте записано слово . Если значение  определено, то конечного числа шагов (тактов) машина должна перейти в заключительную конфигурацию, в которой на ленте записано слово . В противном случае МТ должна работать бесконечно.

Числовая функция  **правильно вычислима** (или просто вычислима) **по Тьюрингу**, если существует МТ, которая переводит конфигурацию в конфигурацию , когда =*y* , или работает бесконечно, когда  не определена.

**Задание на лабораторную работу**

1.Описать системой команд, функциональной таблицей и диаграммой переходов работу машины Тьюринга, реализующую заданный вариантом алгоритм. Начальная и конечная конфигурации стандартны.

2. Проверить модель алгоритма на множестве тестовых примеров. Привести последовательности конфигураций машины Тьюринга, заданной в предыдущем пункте, для различных тестовых исходных слов.

**Варианты заданий**

1. Реализовать функцию арифметическое вычитание в унарном коде.

2. Реализовать функцию выбор максимального из двух чисел  над числами в унарном коде.

3. Реализовать функцию  над числами в унарном коде.

4. Реализовать функцию  над числами в унарном коде.

5. Реализовать функцию  над числами в унарном коде.

6. Реализовать функцию  над числами в унарном коде.

7. Реализовать функцию выбор аргумента  над числами в унарном коде.

8. Реализовать вычисление предиката *X>Y* в унарном коде с сохранением (восстановлением) исходных данных.

9. Реализовать вычисление предиката *X=Y* в унарном коде с сохранением (восстановлением) исходных данных.

10. Реализовать вычисление предиката “x - четное число” в двоичном коде.

11. Реализовать алгоритм в алфавите , меняющий местами первую и последнюю буквы слова.

12. Реализовать алгоритм над алфавитом , меняющий местами первый ноль и последнюю единицу.

13. Реализовать операцию копирование в алфавите , то есть получить из слова  слово .

14. Реализовать алгоритм над алфавитом , который выдает единицу, если в исходном слове только парные нули и ноль в противном случае.

15. Реализовать алгоритм в алфавите , который переставляет буквы в слове  так, чтобы сначала шли все нули, потом – единицы.

16. Реализовать алгоритм, конструирующий в алфавите  слова вида , где - произвольное натуральное число.

17. Реализовать алгоритм, реализующий функцию циклический сдвиг двоичного числа на одну ячейку.

18. Реализовать алгоритм в алфавите , анализирующий последовательность цифр в слове и выдающий «+», если цифры образуют неубывающую последовательность, и «–» в противном случае.

19. Реализовать выделение подстроки, заключенной между двумя символами  (первая пара) в алфавите . Если последовательность  отсутствует на ленте, стереть все.

20. В слове  в алфавите  стереть все, кроме . Если такой последовательности нет, все стереть.

21. Реализовать алгоритм над алфавитом , переставляющий буквы в обратном порядке.

22. Реализовать предиката «в слове α в алфавите есть пара букв ‘*yy*’ » .

23. Реализовать алгоритм в алфавите , производящий в слове α алфавита замену всех вхождений буквы *а* на букву *б*.

24. Реализовать алгоритм в алфавите для вычисления логической функции , где *x,y,z* принимают значение 0 или 1.

25. Реализовать алгоритм в алфавите для вычисления логической функции , где *x,y,z* принимают значение 0 или 1.

**Контрольные вопросы**

1. Дать определение машины Тьюринга (МТ) и ее составляющим.
2. Перечислить и определить способы описания МТ.
3. Какие операции выполняются в каждом такте работы МТ?
4. Дать определение конфигурации МТ.
5. Какие начальные и конечные конфигурации называют стандартными и как они обозначаются?
6. Что такое функция, правильно вычислимая по Тьюрингу?
7. Какие способы композиции МТ существуют, как они применяются и обозначаются?
8. Формулировка тезиса Тьюринга; можно ли его доказать строго?