

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Локтионова Оксана Геннадьевна
Должность: проректор по учебной работе
Дата подписания: 17.05.2024 07:49:20
Уникальный программный ключ:
0b817ca911e6668abb13a5d426d39e5f1c11eabbf73e943df4a4851fda56d089

МИНОБРАЗОВАНИЯ РОССИИ
Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра вычислительной техники



Машина Тьюринга

Методические указания
по выполнению лабораторной работы по дисциплине
«Организация систем искусственного интеллекта
для студентов направления подготовки 09.03.01
Информатика и вычислительная техника

Курск 2023 г.

УДК 004.89

Составитель Е.Н. Иванова

Рецензент

доцент кафедры программной инженерии,
кандидат технических наук

Т.Н. Конаныхина

Машина Тьюринга: методические указания по выполнению лабораторной работы / Юго-Зап. гос. ун-т; сост.: Е.Н. Иванова. – Курск, 2023. – 17 с.: Библиограф.: с. 17.

Рассматриваются задачи на составление алгоритмов для машины Тьюринга. Приводится краткое описание машины, на примерах объясняются основные приемы составления алгоритмов и предлагаются задачи для решения на занятии.

Методические указания соответствуют требованиям программы, утвержденной учебно-методическим объединением по направлению Информатика и вычислительная техника.

Предназначены для студентов направления 09.03.01 Информатика и вычислительная техника.

Текст печатается в авторской редакции

Подписано в печать . Формат 60x84 1/16.
Усл.печ.л. Уч.-изд.л. . Тираж 20 экз. Заказ . Бесплатно.
Юго-Западный государственный университет.
305040, г. Курск, ул. 50 лет Октября, 94.

1. Цель

Получить представление о машине Тьюринга. Научиться программировать и представлять различными способами работу машины Тьюринга.

2. Описание машины Тьюринга

Машина Тьюринга — простая и полезная абстрактная модель вычислений (компьютерных и цифровых), которая является достаточно общей для воплощения любой компьютерной задачи. Благодаря простому описанию и проведению математического анализа она образует фундамент теоретической информатики. Это исследование привело к более глубокому познанию цифровых компьютеров и исчислений, включая понимание того, что существуют некоторые вычислительные проблемы, не решаемые на общих пользовательских ЭВМ.

Машина Тьюринга (МТ) состоит из:

- а) управляющего устройства, которое может находиться в одном из фиксированного множества внутренних состояний $Q = (q_0, q_1, \dots, q_n)$. Среди элементов множества Q выделены два подмножества: начальное $Q_B = \{q_0\}$ и заключительное $Q_E = \{q_k\}, k = \overline{1, m}$. В начальном состоянии машина находится перед началом работы, перейдя в заключительное состояние, машина останавливается;
- б) бесконечной ленты, разбитой на ячейки; в каждой ячейке может быть записан один символ внешнего алфавита $A = \{a_1, \dots, a_m\}$; один из символов внешнего алфавита называется пустым (для обозначения будем использовать пробел — „_“); в каждый момент на ленте записано конечное число непустых символов. Использование пустого символа удобно тем, что операцию стирания символа в некоторой клетке можно рассматривать как запись в эту

клетку пустого символа („_“), поэтому вместо длинной фразы „записать символ в клетку или стереть находящийся там символ“ можно говорить просто „записать символ в клетку“;

- в) считывающей и записывающей головки, которая обзревает одну ячейку ленты, записывает в нее новый символ из внешнего алфавита (он может совпадать с прежним), сдвигается влево или вправо на одну ячейку или остается на месте. Управляющее устройство при этом может изменить свое внутреннее состояние.

Активную часть МТ (управляющее устройство и считывающая/записывающая головка) можно рассматривать как автомат. В каждый момент он размещается под одной из клеток ленты и видит её содержимое; это видимая клетка, а находящийся в ней символ — видимый символ; содержимое соседних и других клеток автомат не видит. Кроме того, в каждый момент автомат находится в одном из состояний: q_1 , q_2 и т.п. Находясь в некотором состоянии, автомат выполняет какую-то определённую операцию (например, перемещается направо по ленте, заменяя все символы b на a), находясь в другом состоянии — другую операцию.

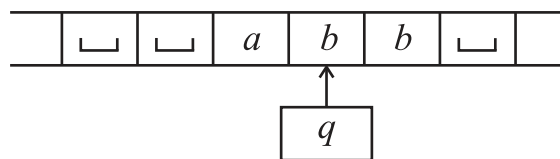


Рисунок 1 – Представление машины Тьюринга

Управляющее устройство будем обозначать прямоугольником, в котором записано внутреннее состояние, а обзреваемую ячейку стрелкой так, как это указано на рис. 1. Машина Тьюринга работает детерминировано, тактами. Такт состоит в выполнении одной команды — выражения вида $q_i a_j \rightarrow q_k a_l d_m$, где $d_m \in \{L, R, H\}$. Эта команда содержит условие (левая часть команды до стрелки) и дей-

ствие (правая часть команды после стрелки). Команда выполняется следующим образом. Если на начало очередного такта внутреннее состояние машины q_i , обозреваемый символ a_j , то после выполнения команды $q_i a_j \rightarrow q_k a_l d_m$, т.е. к концу такта, внутреннее состояние становится q_k , в обозреваемую ячейку записывается символ a_l , и головка сдвигается на одну ячейку вправо, если $d_m = R$, сдвигается влево, если $d_m = L$, остается на месте, если $d_m = H$.

Пару из видимого символа (S) и текущего состояния автомата (q) будем называть конфигурацией и обозначать $\langle S, q \rangle$. Автомат может выполнять три элементарных действия:

- 1) записывать в видимую клетку новый символ (менять содержимое других клеток автомат не может, в частности, может быть записан тот же символ, что и был в ней, тогда содержимое этой клетки не меняется);
- 2) сдвигаться на одну клетку влево (обозначение — L , от „left“) или вправо (обозначение — R , от „right“) („перепрыгивать“ сразу через несколько клеток автомат не может), либо остается неподвижным (обозначение — H);
- 3) переходить в новое состояние (в частности, может остаться в прежнем состоянии).

Ничего другого делать автомат не умеет, поэтому все более сложные операции так или иначе должны быть сведены к этим трём элементарным действиям. Машина начинает работу, обозревая самую левую непустую ячейку и находясь в начальном внутреннем состоянии.

Работа машины Тьюринга не всегда заканчивается. Рассмотрим примеры. Предположим, что программа машины содержит следующие команды

$$q_1 a \rightarrow q_2 a R, q_2 b \rightarrow q_1 b L, q_3 _ \rightarrow q_3 _ R.$$

Если машина на начало очередного такта находится в состоянии q_1 , обозревает ячейку, в которой записан символ a , а в

следующей записан символ b , то произойдет заикливание на участке ленты, содержащем эти символы, и машина будет работать бесконечно. Заикливание работы машины может произойти еще в одном случае, если машина находится в состоянии q_3 , обзореваает пустую ячейку (с символом $_$) и все следующие ячейки так же пусты, то машина также заикливается, но уже на бесконечном участке ленты.

Сама по себе МТ ничего не делает. Для того, чтобы заставить её работать, надо написать для неё программу. Программа — это множество команд с различными условиями, т.е. различными левыми частями. Программа записывается в виде следующей таблицы: (табл. 1).

Таблица 1 – Алгоритм работы машины Тьюринга

	q_0	q_1	\dots	q_n
s_1			\dots	
s_2			\dots	
\dots	\dots	\dots	\dots	\dots
s_i		$s_j[L, R, H]q_k$		
\dots	\dots	\dots	\dots	\dots
s_n			\dots	
$_$			\dots	

Сверху перечисляются все состояния, в которых может находиться автомат, слева — все символы (в том числе и $_$), которые автомат может видеть на ленте. (Какие именно символы и состояния указывать в таблице — определяет автор программы.) На пересечениях же (в ячейках таблицы) указываются те такты, которые должен выполнить автомат, когда он находится в соответствующем состоянии и видит на ленте соответствующий символ. В целом таблица определяет действия МТ при всех возможных конфигурациях и тем самым полностью задаёт поведение МТ. Описать алгоритм в виде МТ — значит предъявить такую таблицу.

До выполнения программы нужно проделать следующие предварительные действия. Во-первых, надо записать на ленту входное слово, к которому будет применена программа. Входное слово — это конечная последовательность символов, записанных в соседних клетках ленты; внутри входного слова пустых клеток быть не должно, а слева и справа от него должны быть только пустые клетки. Пустое входное слово означает, что все клетки ленты пусты. Во-вторых, надо установить автомат в состояние q_0 (указанное в таблице первым) и разместить его под первым символом входного слова. Если входное слово пустое, то автомат может смотреть в любую клетку, т.к. все они пусты. После этих предварительных действий начинается выполнение программы. В таблице отыскивается ячейка на пересечении первого столбца (т.к. автомат находится в состоянии q_0) и той строки, который соответствует первому символу входного слова (это необязательно верхняя строка таблицы), и выполняется такт, указанный в этой ячейке. В результате автомат окажется в новой конфигурации. Теперь такие же действия повторяются, но уже для новой конфигурации: в таблице отыскивается ячейка, соответствующая состоянию и символу этой конфигурации, и выполняется такт из этой ячейки. И так далее.

Когда завершается выполнение программы? Введём понятие такта останова. Это такт, который ничего не меняет: автомат записывает в видимую клетку тот же символ, что и был в ней раньше, не сдвигается и остается в прежнем состоянии. Попав на такт останова, МТ, по определению, останавливается, завершая свою работу.

В целом возможны два исхода работы МТ над входным словом:

1) Первый исход — „хороший“: это когда в какой-то момент МТ останавливается (попадает на такт останова). В таком случае говорят, что МТ применима к заданному входному

слову. А то слово, которое к этому моменту получено на ленте, считается выходным словом, т.е. результатом работы МТ, ответом. В момент останова должны быть выполнены следующие обязательные условия:

— внутри выходного слова не должно быть пустых клеток (отметим, что во время выполнения программы внутри обрабатываемого слова пустые клетки могут быть, но в конце их уже не должно остаться);

— автомат обязан остановиться под одним из символов выходного слова (под каким именно — не играет роли), а если слово пустое — под любой клеткой ленты.

2) Второй исход — „плохой“: это когда МТ зацикливается, никогда не попадая на такт останова (например, автомат на каждом шаге сдвигается вправо и потому не может остановиться, т.к. лента бесконечна). В этом случае говорят, что МТ неприменима к заданному входному слову. Ни о каком результате при таком исходе не может идти и речи. Один и тот же алгоритм (программа МТ) может быть применимым к одним входным словам (т.е. останавливаться) и неприменимым к другим (т.е. зацикливаться). Таким образом, применимость/неприменимость зависит не только от самого алгоритма, но и от входного слова. На каких входных словах алгоритм должен останавливаться? На тех, которые относятся к допустимым исходным данным решаемой задачи, для которых задача осмысленна. Но на ленте могут быть записаны любые входные слова, в том числе и те, для которых задача не имеет смысла; на таких словах поведение алгоритма не фиксируется, он может остановиться (при любом результате), а может и зациклиться.

Еще один способ описания программы МТ — описание с помощью диаграмм. Диаграмма представляет собой ориентированный граф с помеченными вершинами и дугами и имеет вид, изображенный на рисунке 2. Вершины диаграм-

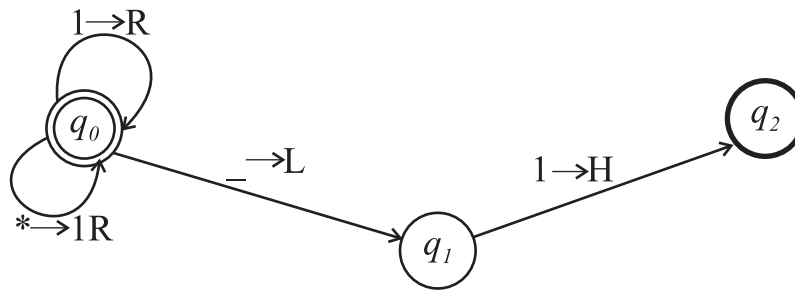


Рисунок 2 – Представление МТ с помощью диаграммы

мы представляют собой внутренние состояния. Если машина имеет команду $q_i a_j \rightarrow q_k a_l d_m$, то от вершины q_i к вершине q_k проводится дуга, которая помечается выражением $a_j \rightarrow a_l d_m$, если $a_j \rightarrow a_l$, и выражением $a_j \rightarrow d_m$, если $a_j = a_l$. Начальное состояние на диаграмме обозначается „двойным“ кружком, а заключительное — „жирным“ кружком.

3. Примеры программирования машины Тьюринга

Рассмотрим примеры на составление программ для МТ, чтобы продемонстрировать некоторые типичные приёмы программирования на МТ. Для сокращения формулировки задач введём следующие два соглашения:

- буквой P будем обозначать входное слово;
- буквой A будем обозначать алфавит машины Тьюринга, т.е. набор тех символов, из которых может состоять P .

Пример 1 (перемещение автомата, замена символов)

Алфавит $A = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$. Пусть P — непустое слово, значит, P — это последовательность из десятичных цифр, т.е. запись неотрицательного целого числа в десятичной системе. Требуется получить на ленте запись числа, которое на 1 больше числа P .

Решение. Для решения этой задачи предлагается выполнить следующие действия:

1. Перегнать автомат под последнюю цифру числа.
2. Если это цифра от 0 до 8, то заменить её цифрой на 1

больше и остановиться (рис. 3).

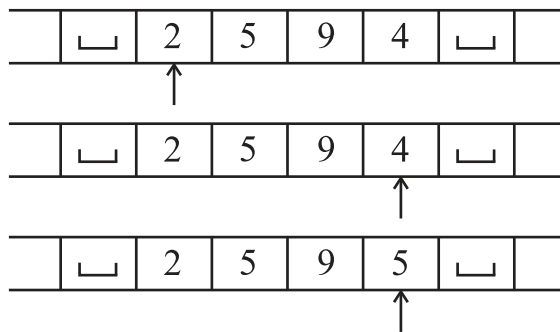


Рисунок 3 – Выполнение инкремента для цифр от 0 до 8

3. Если же это цифра 9, тогда заменить её на 0 и сдвинуть автомат к предыдущей цифре, после чего таким же способом увеличить на 1 эту предпоследнюю цифру (рис. 4).

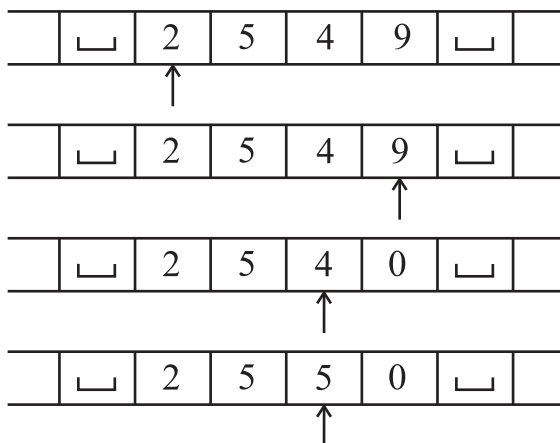


Рисунок 4 – Выполнение инкремента для цифры 9

4. Особый случай: в P только девятки (например, 99). Тогда автомат будет сдвигаться влево, заменяя девятки на нули, и в конце концов окажется под пустой клеткой. В эту пустую клетку надо записать 1 и остановиться (ответом будет 100) (рис. 5).

В виде программы для МТ эти действия описываются следующим образом (таблица 2).

Пояснения к таблице 2.

q_0 — это состояние, в котором автомат „бежит“ под последнюю цифру числа. Для этого он всё время движется вправо, не меняя видимые цифры и оставаясь в том же

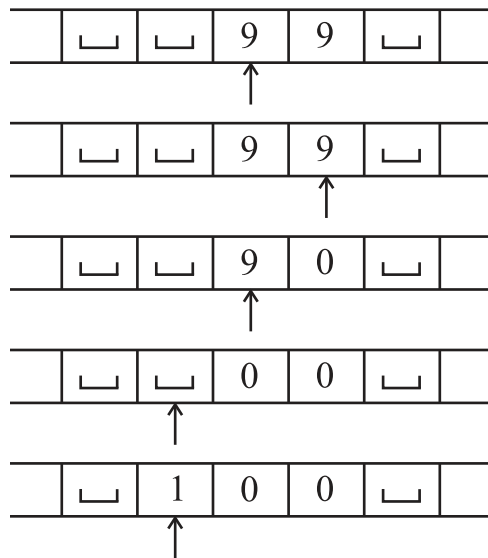


Рисунок 5 – Выполнение инкремента для числа 99

состоянии. Но здесь есть одна особенность: когда автомат находится под последней цифрой, то он ещё не знает об этом (ведь он не видит, что записано в соседних клетках) и определит это лишь тогда, когда попадёт на пустую клетку. Поэтому, дойдя до первой пустой клетки, автомат возвращается назад под последнюю цифру и переходит в состояние q_1 (вправо двигаться уже не надо).

q_1 — это состояние, в котором автомат прибавляет 1 к той цифре, которую видит в данный момент. Сначала это последняя цифра числа. Если она находится в диапазоне от 0 до 8, то автомат заменяет её цифрой, которая на 1 больше, и останавливается. Но если это цифра 9, то автомат заменяет её на 0 и сдвигается влево, оставаясь в состоянии q_1 . Тем самым, он будет теперь прибавлять 1 к предыдущей цифре. Если и эта цифра равна 9, то автомат заменяет её на 0 и сдвигается влево, оставаясь по-прежнему в состоянии q_1 , т.к. должен выполнить то же самое действие — увеличить на 1 видимую цифру. Если же автомат сдвинулся влево, а в видимой клетке нет цифры (а есть „пусто“), то он записывает сюда 1 и останавливается. Отметим, что для пустого входного слова наша задача не определена, поэтому на этом

Таблица 2 – Операция инкремента

	q_0	q_1
0	0, R, q_0	1, H, q_k
1	1, R, q_0	2, H, q_k
2	2, R, q_0	3, H, q_k
3	3, R, q_0	4, H, q_k
4	4, R, q_0	5, H, q_k
5	5, R, q_0	6, H, q_k
6	6, R, q_0	7, H, q_k
7	7, R, q_0	8, H, q_k
8	8, R, q_0	9, H, q_k
9	9, R, q_0	0, L, q_1
–	–, L, q_1	1, H, q_k

слове МТ может вести себя как угодно. В нашей программе, например, при пустом входном слове МТ останавливается и выдает ответ 1.

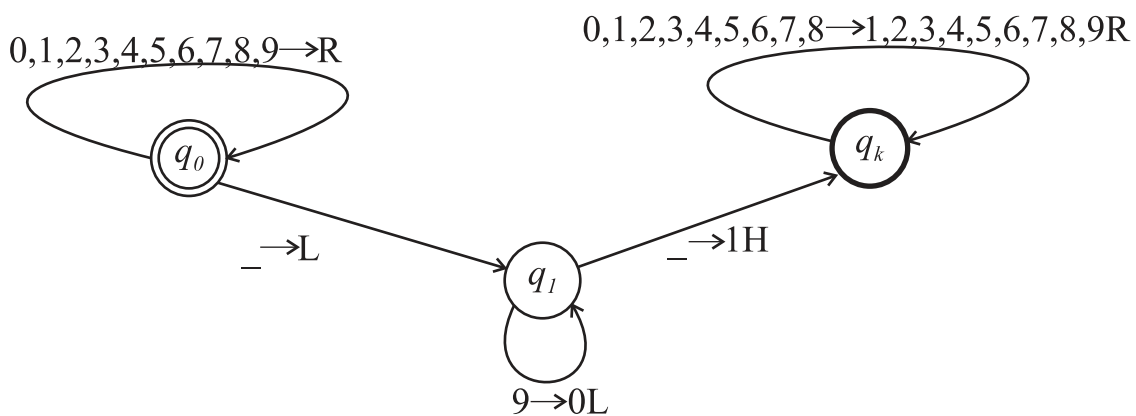


Рисунок 6 – Диаграмма МТ для операции инкремента

Диаграмма МТ, выполняющей операцию инкремента целого десятичного числа, приведена на рисунке 6.

Пример 2 (сравнение символов, стирание слова)

Алфавит $A = \{a, b, c\}$. Если первый и последний символы (непустого) слова P одинаковы, тогда это слово не ме-

нять, а иначе заменить его на пустое слово.

Решение. Для решения этой задачи предлагается выполнить следующие действия:

1. Запомнить первый символ входного слова, не стирая его.
2. Переместить автомат под последний символ и сравнить его с запомненным. Если они равны, то больше ничего не делать.
3. В противном случае уничтожить всё входное слово.

Как запоминать символ и как перегонять автомат под последний символ слова, рассмотрено в примере 1. Стирание же входного слова реализуется заменой всех его символов на символ „_“. При этом, раз уж автомат оказался в конце слова, будем перемещать автомат справа налево до первой пустой клетки. Эти действия описываются программой для МТ, приведенной в таблице 3.

Таблица 3 – Сравнение символов, стирание слова

	q_0	q_1	q_2	q_3	q_4	q_5	q_6	q_7
a	aHq_1	aRq_1	aHq_8	aRq_3	aHq_7	aRq_5	aHq_7	$_Lq_7$
b	bHq_3	bRq_1	bHq_7	bRq_3	bHq_8	bRq_5	bHq_7	$_Lq_7$
c	cHq_5	cRq_1	cHq_7	cRq_3	cHq_7	cRq_5	cHq_8	$_Lq_7$
$_$	$_Hq_8$	$_Lq_2$	\times	$_Lq_4$	\times	$_Lq_6$	\times	$_Hq_8$

Пояснения к таблице 3.

Крестик в ячейке таблицы означает невозможность появления соответствующей конфигурации при выполнении программы.

q_0 — состояние, в котором анализируется 1-ый символ и происходит разветвление.

q_1 — состояние, в котором автомат идет к последнему символу при 1-м символе a .

q_2 — состояние, в котором сравнивается последний символ с a , если символы равны, то автомат заканчивает

свою работу, иначе переходит к стиранию записанного на ленте слова.

q_3, q_4 — состояния, аналогичные q_1, q_2 при 1-м символе b .

q_5, q_6 — состояния, аналогичные q_1, q_2 при 1-м символе c .

q_7 — состояние стирания всего слова, двигаясь справа налево.

4. Задания для самостоятельного выполнения

Составить программу МТ в виде таблицы и диаграммы. В задачах рассматриваются только целые неотрицательные числа, если не указано иное. Под „единичной“ системой счисления понимается запись неотрицательного целого числа помощью палочек — должно быть выписано столько палочек, какова величина числа, например: $2 \rightarrow ||$, $5 \rightarrow |||||$.

1. $A = \{a, b, c, _ \}$. Приписать слева к слову P символ b ($P \rightarrow bP$).
2. $A = \{a, b, c, _ \}$. Приписать справа к слову P символы bc ($P \rightarrow Pbc$).
3. $A = \{a, b, c, _ \}$. Заменить на a каждый второй символ в слове P .
4. $A = \{a, b, c, _ \}$. Оставить в слове P только первый символ (пустое слово не менять).
5. $A = \{a, b, c, _ \}$. Оставить в слове P только последний символ (пустое слово не менять).
6. $A = \{a, b, c, _ \}$. Определить, является ли P словом ab . Ответ (выходное слово): слово ab , если является, или пустое слово иначе.
7. $A = \{a, b, c, _ \}$. Определить, входит ли в слово P символ a . Ответ: слово из одного символа a (да, входит) или пустое слово (нет, не входит).

8. $A = \{a, b, c, _ \}$. Если в слово P не входит символ a , то заменить в P все символы b на c , иначе в качестве ответа выдать слово из одного символа a .
9. $A = \{a, b, 0, 1, _ \}$. Определить, является ли слово P идентификатором (непустым словом, начинающимся с буквы). Ответ: слово a („да“) или пустое слово („нет“).
10. $A = \{a, b, 0, 1, _ \}$. Определить, является ли слово P записью числа в двоичной системе счисления (непустым словом, состоящем только из цифр 0 и 1). Ответ: слово 1 („да“) или слово 0 („нет“).
11. $A = \{0, 1, _ \}$. Считая непустое слово P записью двоичного числа, удалить из него незначащие нули, если такие есть.
12. $A = \{0, 1, _ \}$. Для непустого слова P определить, является ли оно записью степени двойки (1, 2, 4, 8, ...) в двоичной системе счисления. Ответ: слово 1 („является“) или слово 0 („не является“).
13. $A = \{0, 1, 2, 3, _ \}$. Считая непустое слово P записью числа в четверичной системе счисления, определить, является оно чётным числом или нет. Ответ: 1 („да“) или 0 („нет“).
14. $A = \{0, 1, _ \}$. Считая непустое слово P записью числа в двоичной системе, получить двоичное число, равное учетверенному числу P (например: $101 \rightarrow 10100$).
15. $A = \{0, 1, _ \}$. Считая непустое слово P записью числа в двоичной системе, получить двоичное число, равное неполному частному от деления числа P на 2 (например: $1011 \rightarrow 101$).
16. $A = \{a, b, c, _ \}$. Если P — слово чётной длины (0, 2, 4, ...), то выдать ответ a , иначе — пустое слово.
17. $A = \{0, 1, 2, _ \}$. Считая непустое слово P записью числа в троичной системе счисления, определить, является оно чётным числом или нет. Ответ: 1 („да“) или 0 („нет“).
Замечание: в чётном троичном числе должно быть чётное количество цифр 1.

18. $A = \{a, b, c, _ \}$. Пусть P имеет нечётную длину. Оставить в P только средний символ.
19. $A = \{a, b, c, _ \}$. Если слово P имеет чётную длину, то оставить в нём только левую половину.
20. $A = \{a, b, c, _ \}$. Приписать слева к непустому слову P его первый символ.
21. $A = \{a, b, _ \}$. Для непустого слова P определить, входит ли в него ещё раз его первый символ. Ответ: a („да“) или пустое слово.
22. $A = \{a, b, _ \}$. В непустом слове P поменять местами его первый и последний символы.
23. $A = \{a, b, _ \}$. Определить, является P палиндромом (перевёртышем, симметричным словом) или нет. Ответ: a („да“) или пустое слово.
24. $A = \{a, b, _ \}$. Заменить в P каждое вхождение a на bb .
25. $A = \{a, b, c, _ \}$. Заменить в P каждое вхождение ab на c .
26. $A = \{a, b, _ \}$. Удвоить слово P (например: $abb \rightarrow abbabb$).
27. $A = \{a, b, _ \}$. Удвоить каждый символ слова P (например: $bab \rightarrow bbaabb$).
28. $A = \{a, b, _ \}$. Перевернуть слово P (например: $abb \rightarrow bba$).
29. $A = \{0, 1, _ \}$. Считая непустое слово P записью двоичного числа, получить это же число, но в четверичной системе. Замечание: учесть, что в двоичном числе может быть нечётное количество цифр.
30. $A = \{0, 1, 2, 3, _ \}$. Считая непустое слово P записью числа в четверичной системе счисления, получить запись этого числа в двоичной системе.
31. $A = \{0, 1, 2, _ \}$. Считая непустое слово P записью положительного числа в троичной системе счисления, уменьшить это число на 1.
32. $A = \{|_ \}$. Считая слово P записью числа в единичной системе счисления, получить запись этого числа в троичной системе. Рекомендация: следует в цикле удалять из

„единичного“ числа по палочке и каждый раз прибавлять 1 к троичному числу, которое вначале положить равным 0.

33. $A = \{0, 1, 2, _ \}$. Считая непустое слово P записью числа в троичной системе счисления, получить запись этого числа в единичной системе.

5. Контрольные вопросы

1. Обоснуйте свойства машины Тьюринга как алгоритма.
2. Что такое алфавит машины Тьюринга?
3. Каковы составные части машины Тьюринга?
4. Что представляет собой программа функционирования машины Тьюринга?
5. Какие операции может выполнять автомат машины Тьюринга?
6. Что такое входное слово?
7. Какими могут быть результаты работы машины Тьюринга?
8. Как обозначаются начальное и конечные состояния на диаграмме, представляющей машину Тьюринга?

6. Литература

1. Сидоркина, И.Г. Системы искусственного интеллекта [Текст]: учебное пособие / И.Г. Сидоркина. — Москва: КНОРУС, 2016. — 246 с.
2. Системы искусственного интеллекта. Практический курс [Текст]: учебное пособие / ред. И.Ф. Астахова. — М.: БИНОМ. Лаборатория знаний, 2008. — 292 с.