Lab09-Turing Machine and Reduction

Algorithm and Complexity (CS214), Xiaofeng Gao, Spring 2018.

∗ If there is any problem, please contact TA Xinyu Wu.
∗ Name:Juncheng Wan Student ID:516021910620 Email: 578177149@qq.com

1. Design an one-tape TM M that compute the function $f(x) = x \mod y$, says, the reminder of dividing x by y, where x and y belong to the natural number set $\mathbb{N}$. The alphabet is $\{1, 0, \square, \triangleright, \triangleleft\}$, where the input on the tape are x "1"'s, $\square$ and y "1"s. Below is the initial configurations for input x and y. The result is the number of "1"s on the tape with pattern of $\triangleright 111\cdots 111\triangleleft$. First describe your design and then write the specifications of M in the form like $\langle q_S, \triangleright\rangle \rightarrow \langle q_1, \triangleright, R\rangle$ and then explain the transition functions in detail. Moreover, you should draw the state transition diagram. (You can get bonus if you write briefly and clearly the whole process from initial to final configurations when $x = 7$ and $y = 3$.)

   Solution. I answer this question step by step as follows.

   (a) **Description:**

   First, I let the reading head move right to the first "1" after $\square$. The state will change from q_S to q_R.

   Second, at the state of q_R, the reading head will first move left to set the first "1" before $\square$ to "0", and move right to set the first "1" after $\square$ to "0", too. If $x > y$, the reading head will change the state from q_R to q_L. At the state of q_L, each time the first "0" before $\triangleright$ will transform to "1" with the first "1" before $\square$ transforming to 0. If all the "0"s between $\square$ and $\triangleleft$ transform to "1"s, the state will transform from q_L to a_R and repeat the above steps.

   Third, after all the "1"s between $\triangleright$ and $\square$ transform to "0"s, the following step is up to the state q. If $q = q_L$, all the "0"s between $\square$ and $\triangleleft$ will transform to "1"s with the "0" the nearest to $\triangleleft$ transforming to "1"s. If $q = q_R$, all the "1"s between $\square$ and $\triangleleft$ will transform to "0"s with the "0" the nearest to $\triangleleft$ transforming to "1"s.

   After this, the state will transform from q_L or q_R to q_f. At the q_f state, the cell next to the rightest "1" between $\triangleleft$ and $\square$ will transform to $\triangleleft$. Thus, there will be two $\triangleleft$ and just set the cell in this two $\triangleleft$ $\square$ and the second $\triangleleft$ to $\square$ too.

   Finally, the state transforms from q_F to q_H.

   (b) **The specifications of M:**

   $\langle q_S, \triangleright\rangle \rightarrow \langle q_{R1}, \triangleright, R\rangle$
   $\langle q_{R1}, 1\rangle \rightarrow \langle q_{R1}, 1, R\rangle$
   $\langle q_{R1}, \square\rangle \rightarrow \langle q_{R2}, \square, R\rangle$
   $\langle q_{R2}, 0\rangle \rightarrow \langle q_{R2}, 0, R\rangle$
   $\langle q_{R2}, 1\rangle \rightarrow \langle q_{R3}, 0, L\rangle$
   $\langle q_{R3}, 0\rangle \rightarrow \langle q_{R3}, 0, L\rangle$
   $\langle q_{R3}, \square\rangle \rightarrow \langle q_{R4}, \square, L\rangle$
   $\langle q_{R4}, 0\rangle \rightarrow \langle q_{R4}, 0, L\rangle$
   $\langle q_{R4}, 1\rangle \rightarrow \langle q_{R5}, 0, R\rangle$
   $\langle q_{R5}, 0\rangle \rightarrow \langle q_{R5}, 0, R\rangle$
   $\langle q_{R5}, \square\rangle \rightarrow \langle q_{R2}, \square, R\rangle$
   $\langle q_{R2}, \triangleleft\rangle \rightarrow \langle q_{L1}, \triangleleft, L\rangle$

$\langle q_{L1}, 0\rangle \to \langle q_{L1}, 0, L\rangle$
$\langle q_{L1}, \square\rangle \to \langle q_{L2}, \square, L\rangle$
$\langle q_{L2}, 0\rangle \to \langle q_{L2}, 0, L\rangle$
$\langle q_{L2}, 1\rangle \to \langle q_{L3}, 0, R\rangle$
$\langle q_{L3}, 0\rangle \to \langle q_{L3}, 0, R\rangle$
$\langle q_{L3}, \square\rangle \to \langle q_{L4}, \square, R\rangle$
$\langle q_{L4}, 0\rangle \to \langle q_{L5}, 0, R\rangle$
$\langle q_{L5}, 1\rangle \to \langle q_{L5}, 1, R\rangle$
$\langle q_{L5}, \triangleleft\rangle \to \langle q_{L6}, \triangleleft, L\rangle$
$\langle q_{L6}, 1\rangle \to \langle q_{L6}, 1, L\rangle$
$\langle q_{L6}, 0\rangle \to \langle q_{L1}, 1, L\rangle$
$\langle q_{L4}, 1\rangle \to \langle q_{R4}, 0, R\rangle$
$\langle q_{R4}, \triangleright\rangle \to \langle q_{FR1}, \triangleright, R\rangle$
$\langle q_{FR1}, 0\rangle \to \langle q_{FR1}, 0, R\rangle$
$\langle q_{FR1}, \square\rangle \to \langle q_{FR2}, \square, R\rangle$
$\langle q_{FR2}, 1\rangle \to \langle q_{FR2}, 1, R\rangle$
$\langle q_{FR2}, 0\rangle \to \langle q_{FR3}, 1, L\rangle$
$\langle q_{FR3}, 1\rangle \to \langle q_{FR3}, 1, L\rangle$
$\langle q_{FR3}, \square\rangle \to \langle q_{FR4}, \square, L\rangle$
$\langle q_{FR4}, 0\rangle \to \langle q_{FR4}, 0, L\rangle$
$\langle q_{FR4}, \triangleright\rangle \to \langle q_{FR5}, \triangleright, R\rangle$
$\langle q_{FR5}, 1\rangle \to \langle q_{FR5}, 1, R\rangle$
$\langle q_{FR5}, 0\rangle \to \langle q_{FR1}, 1, R\rangle$
$\langle q_{FR2}, \triangleleft\rangle \to \langle q_{FR6}, \square, L\rangle$
$\langle q_{FR6}, 1\rangle \to \langle q_{FR6}, \square, L\rangle$
$\langle q_{FR6}, \square\rangle \to \langle q_{FR7}, \square, L\rangle$
$\langle q_{FR7}, 0\rangle \to \langle q_{FR7}, \square, L\rangle$
$\langle q_{FR7}, 1\rangle \to \langle q_{FR8}, 1, R\rangle$
$\langle q_{FR8}, 0\rangle \to \langle q_{H}, \triangleleft, S\rangle$
$\langle q_{FR8}, \square\rangle \to \langle q_{H}, \triangleleft, S\rangle$
$\langle q_{L2}, \triangleright\rangle \to \langle q_{FL1}, \triangleright, R\rangle$
$\langle q_{FL1}, 0\rangle \to \langle q_{FL1}, 0, R\rangle$
$\langle q_{FL1}, \square\rangle \to \langle q_{FL2}, \square, R\rangle$
$\langle q_{FL2}, 1\rangle \to \langle q_{FL3}, 0, L\rangle$
$\langle q_{FL2}, 0\rangle \to \langle q_{FL2}, 0, R\rangle$
$\langle q_{FL3}, \square\rangle \to \langle q_{FL4}, \square, L\rangle$
$\langle q_{FL4}, 0\rangle \to \langle q_{FL4}, 0, L\rangle$
$\langle q_{FL4}, \triangleright\rangle \to \langle q_{FL5}, \triangleright, R\rangle$
$\langle q_{FL5}, 1\rangle \to \langle q_{FL5}, 1, R\rangle$
$\langle q_{FL5}, 0\rangle \to \langle q_{FL6}, 1, R\rangle$
$\langle q_{FL6}, 0\rangle \to \langle q_{FL1}, 0, R\rangle$
$\langle q_{FL2}, \triangleleft\rangle \to \langle q_{FL7}, \square, R\rangle$
$\langle q_{FL2}, 0\rangle \to \langle q_{FL2}, \square, R\rangle$

$\langle q_{FL2}, \square\rangle \to \langle q_{FL8}, \square, R\rangle$
$\langle q_{FL8}, 0\rangle \to \langle q_{FL8}, \square, R\rangle$
$\langle q_{FL8}, 1\rangle \to \langle q_{FL9}, 1, R\rangle$
$\langle q_{FL9}, 0\rangle \to \langle q_H, \triangleleft, S\rangle$
$\langle q_{FL9}, \square\rangle \to \langle q_H, \triangleleft, S\rangle$

(c) **Modification:**

I find that in the above process I use too many states q and it is difficult to draw the state transition diagram. Therefore, I try to modify my algorithm. Each time all the cells between $\square$ and $\triangleleft$ become 0, the machine can try to transform all of them from 0 to 1. Therefore, there is no need to consider q_L or q_R when reducing the 1 between $\triangleright$ and $\square$ and recovering the remainder 1 between $\triangleright$ and $\square$.

i. Start to module:
$\langle q_S, \triangleright\rangle \to \langle q_{m1}, \triangleright, R\rangle$
$\langle q_{m1}, 1\rangle \to \langle q_{m1}, 1, R\rangle$
$\langle q_{m1}, 0\rangle \to \langle q_{m1}, 0, R\rangle$
$\langle q_{m1}, \square\rangle \to \langle q_{m2}, \square, R\rangle$
$\langle q_{m2}, 0\rangle \to \langle q_{m2}, 0, R\rangle$
$\langle q_{m2}, \square\rangle \to \langle q_{m2}, \square, R\rangle$
$\langle q_{m2}, 1\rangle \to \langle q_{m3}, 0, L\rangle$
$\langle q_{m2}, \square\rangle \to \langle q_{m3}, \square, L\rangle$
$\langle q_{m3}, 0\rangle \to \langle q_{m3}, 0, L\rangle$
$\langle q_{m3}, 1\rangle \to \langle q_{m1}, 0, R\rangle$

ii. Transform all the 0 between $\square$ and $\triangleleft$:
$\langle q_{m2}, \triangleleft\rangle \to \langle q_t, \triangleleft, L\rangle$
$\langle q_t, 0\rangle \to \langle q_t, 1, L\rangle$

iii. Recover the remainder:
$\langle q_t, \square\rangle \to \langle q_{m3}, \square, L\rangle$
$\langle q_{m3}, \triangleright\rangle \to \langle q_{r1}, \triangleright, R\rangle$
$\langle q_{r1}, 0\rangle \to \langle q_{r1}, 0, R\rangle$
$\langle q_{r1}, \square\rangle \to \langle q_{r2}, \square, R\rangle$
$\langle q_{r2}, 1\rangle \to \langle q_{r2}, 1, R\rangle$
$\langle q_{r2}, 0\rangle \to \langle q_{r3}, 1, L\rangle$
$\langle q_{r3}, 1\rangle \to \langle q_{r3}, 1, L\rangle$
$\langle q_{r3}, \square\rangle \to \langle q_{r3}, \square, L\rangle$
$\langle q_{r3}, 0\rangle \to \langle q_{r3}, 0, L\rangle$
$\langle q_{r3}, \triangleright\rangle \to \langle q_{r4}, \triangleright, R\rangle$
$\langle q_{r4}, 1\rangle \to \langle q_{r4}, 1, R\rangle$
$\langle q_{r4}, 0\rangle \to \langle q_{r1}, 1, R\rangle$
$\langle q_{r1}, 0\rangle \to \langle q_{r5}, 1, R\rangle$
$\langle q_{r5}, 0\rangle \to \langle q_{r2}, 0, S\rangle$
$\langle q_{r5}, \square\rangle \to \langle q_{r2}, \square, S\rangle$

iv. Finish by cleaning the unrelated to $\square$
$\langle q_{r2}, \triangleleft\rangle \to \langle q_{f1}, \square, L\rangle$
$\langle q_{f1}, 1\rangle \to \langle q_{f1}, 1, L\rangle$
$\langle q_{f1}, \square\rangle \to \langle q_{f2}, \square, L\rangle$
$\langle q_{f2}, 0\rangle \to \langle q_{f2}, \square, L\rangle$
$\langle q_{f2}, 1\rangle \to \langle q_{f3}, 1, R\rangle$
$\langle q_{f2}, \triangleright\rangle \to \langle q_{f3}, \triangleright, R\rangle$
$\langle q_{f3}, \square\rangle \to \langle q_H, \triangleleft, S\rangle$

In this algorithm, there are only 14 states. I think it is more simple.

(d) **The state transition diagram:**

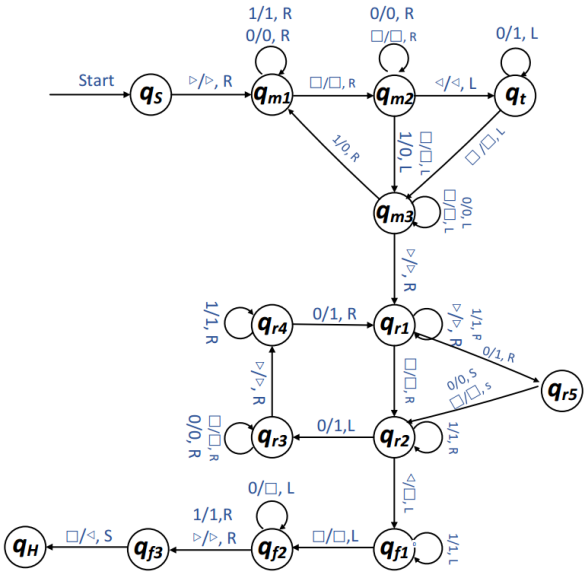
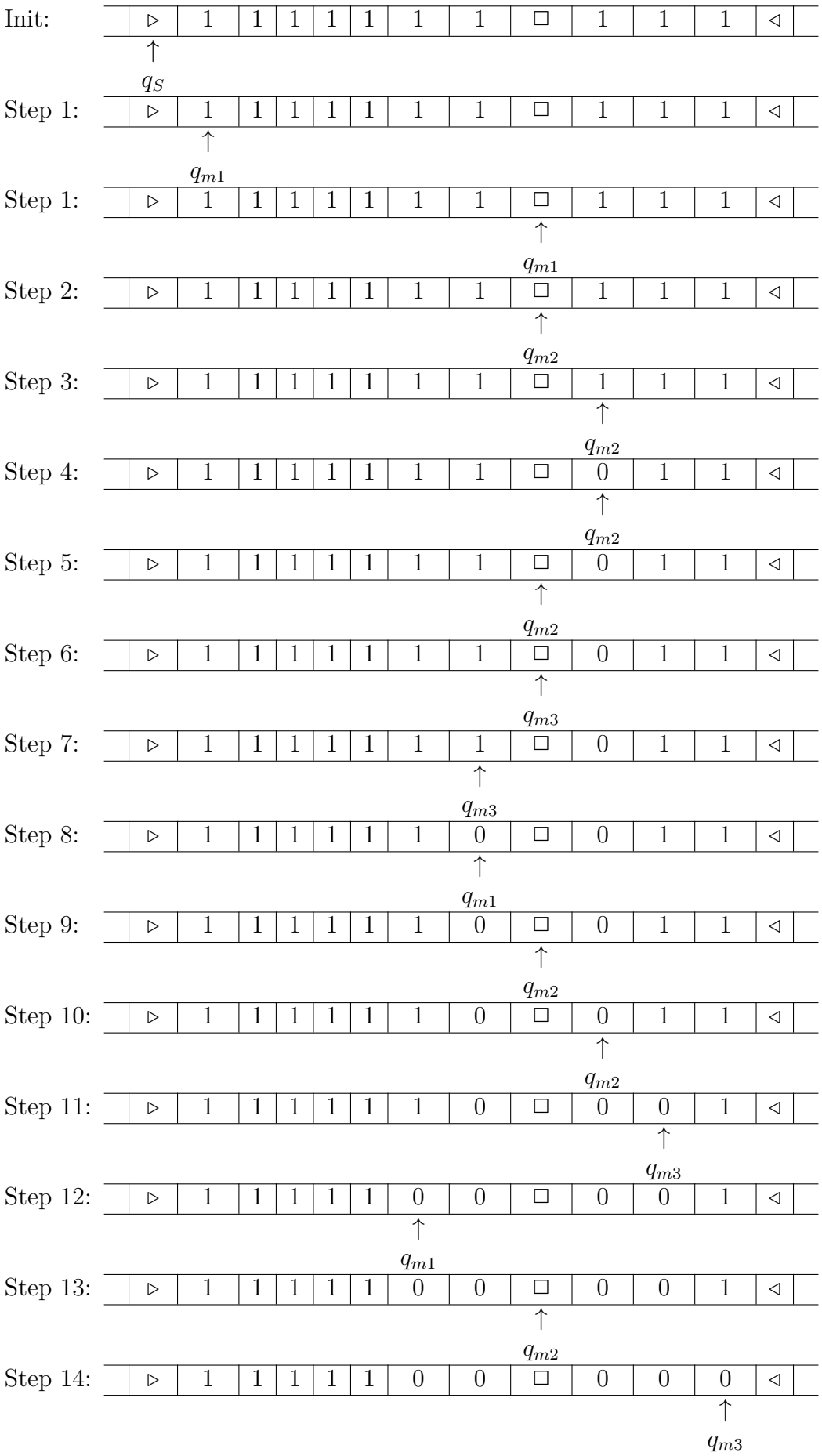


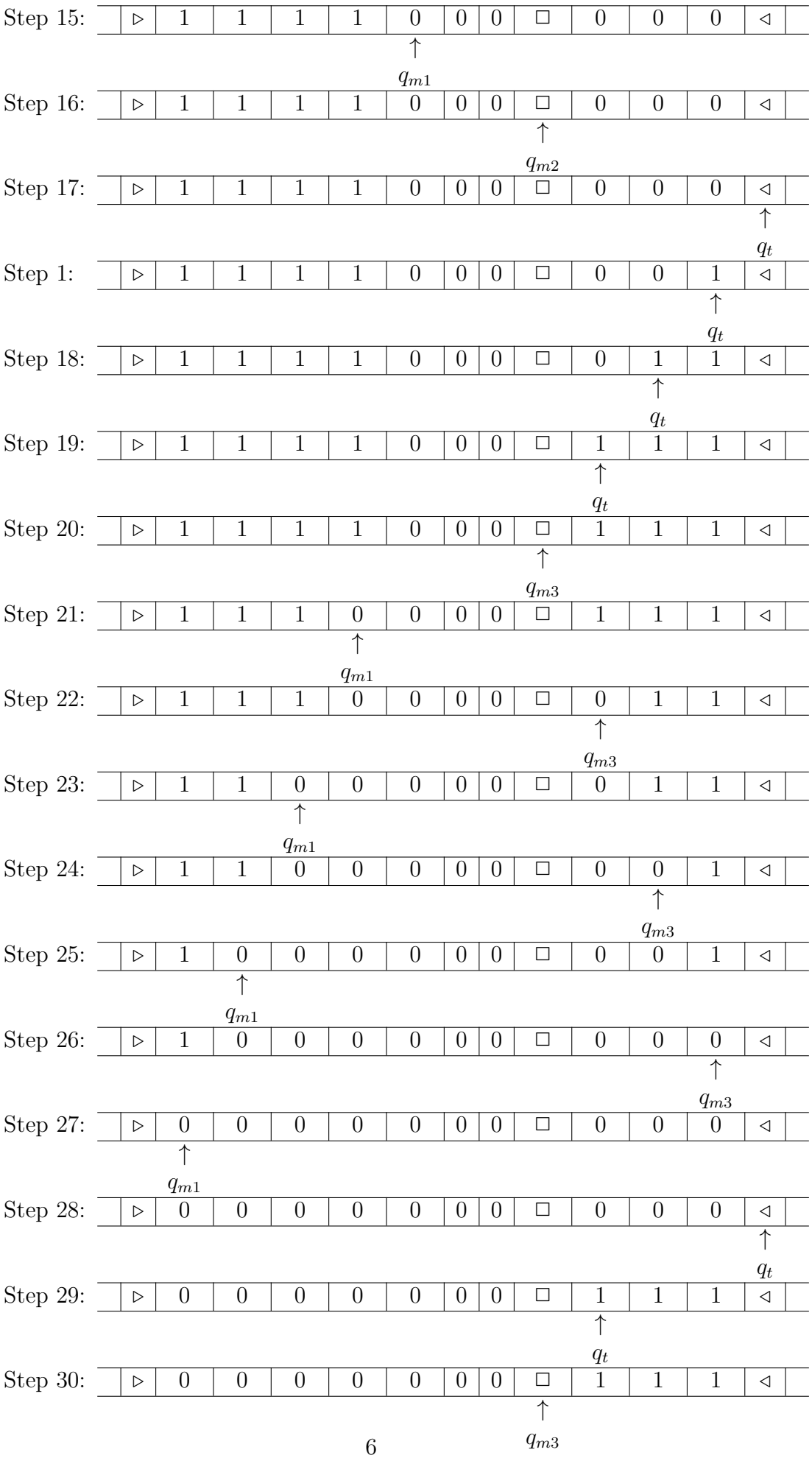
图 1: The state transition diagram.

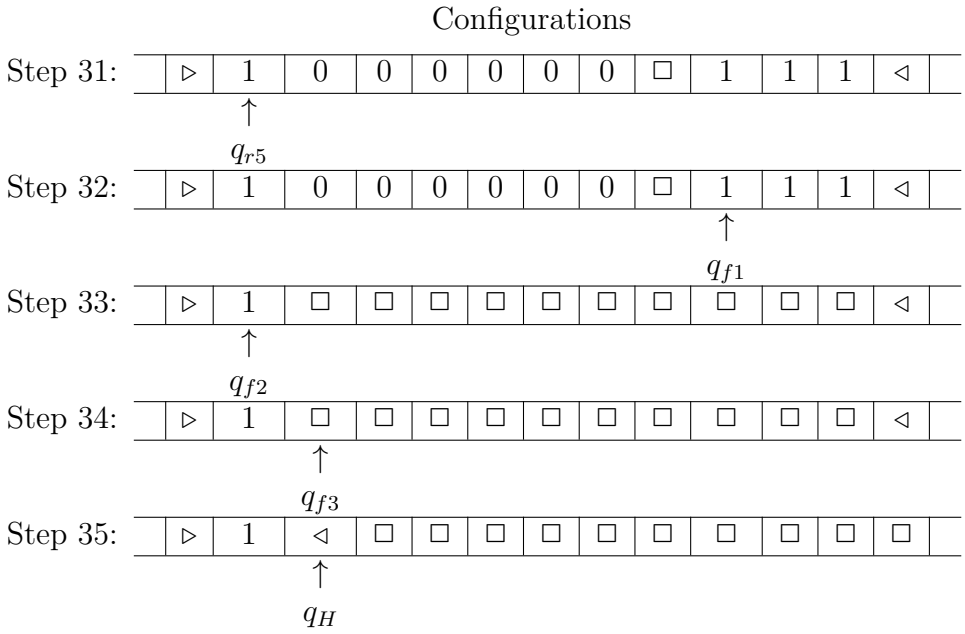
□

Configurations



Configurations





2. $\mathbb{A}$ and $\mathbb{B}$ are two domains other than $\mathbb{N}$. Assume there exist intuitively computable functions $\alpha : \mathbb{A} \to \mathbb{N}$ and $\alpha^{-1} : \mathbb{N} \to \mathbb{A}$ as encoding and decoding functions from $\mathbb{A}$ to $\mathbb{N}$ respectively (we have $\beta : \mathbb{B} \to \mathbb{N}$ and $\beta^{-1} : \mathbb{N} \to \mathbb{B}$ similarly). Then answer the following questions.

(a) To prove that $f : \mathbb{A} \to \mathbb{B}$ is computable, we need to construct an $f^* : \mathbb{N} \to \mathbb{N}$ and analyze its computability. How to calculate f^*?

Solution. To prove that f is conputable, we construct f^* as follows:

$$f^* = \beta \circ f \circ \alpha^{-1} \tag{1}$$

□

(b) Reversely, for a computable function $g : \mathbb{N} \to \mathbb{N}$, we can obtain an intuitively computable function $g' : \mathbb{A} \to \mathbb{B}$ from g. What is g' in notation of g, α and β?

Solution. g' is as follows:

$$g' = \beta^{-1} \circ g \circ \alpha \tag{2}$$

□

(c) Find a new coding function γ (with the help of α and β) to deal with computability on domain $\mathbb{A} \times \mathbb{B}$. (Hint: Consider the "zig-zag" mapping in Figure 2.)

Solution. Set $z(x, y)$ the zig-zag mapping $\mathbb{N} \times \mathbb{N} \to \mathbb{N}$, which is a bijection. Assume that f is a function $f : \mathbb{A} \times \mathbb{B} \to \mathbb{A} \times \mathbb{B}$. Set the new coding function γ to deal with computability on domain $\mathbb{A} \times \mathbb{B}$ as follows:

$$z = z(x, y) = \frac{(x + y)^2 + x + 3y}{2} \tag{3}$$

$$\gamma = z(\alpha, \beta) = \frac{(\alpha + \beta)^2 + \alpha + 3\beta}{2} \Longrightarrow \mathbb{A} \times \mathbb{B} \to \mathbb{N} \tag{4}$$

□

x \ y	0	1	2	3	4
0	0	2	5	9	14
1	1	4	8	13	19
2	3	7	12	18	25
3	6	11	17	24	32
4	10	16	23	31	40

图 2: Zig Zag Mapping

3. What is the "certificate" and "certifier" for the following problems?

(a) *CLIQUE*: Given a graph $G = (V, E)$ and a positive integer k, is there a clique (subsets of vertices which are all adjacent to each other) whose size is no less than k?

Solution. The certificate and the certifier is as follows:

Certificate: a subset of vertices which are adjacent to each other.

Certifier: check that each vertice in the subset is in V (time complexity: $O(|V|)$) and appears only once; check that each vertice is adjacent to other vertices in the subset

(time complexity: $O(|E|)$) ; check that the nunber of vertices is no less than k (time complexity: $O(1)$). Therefore, the time complexity is polynomial. □

(b) *SET PACKING*: Given a finite set U, a positive integer k and several subsets $U_1, U_2, \ldots, U_m$ of U, is there k or more subsets which are disjoint with each other?

Solution. The certificate and the certifier is as follows:
Certificate: a collections $\mathbb{U} = \{U_1, U_2, \ldots, U_m\}$.
Certifier: check that $\forall i \in \{1, 2, \cdots, m\}, U_i \subseteq U$ (time complexity: $O(|U|)$); check that $\forall i \in \{1, 2, \cdots, m\}, U_i$ is disjoint with other subsets (time complexity: $O(m|U|)$); check that $m \geq k$ (time complexity: $O(1)$). Therefore, the time complexity is polynomial. □

(c) *STEINER TREE IN GRAPHS*: Given a graph $G = (V, E)$, a weight $w(e) \in Z_0^+$ for each $e \in E$, a subset $R \subset V$, and a positive integer bound B, is there a subtree of G that includes all the vertices of R and such that the sum of the weights of the edges in the subtree is no more than B.

Solution. The certificate and the certifier is as follows:
Certificate: a graph $T(V_T, E_T)$.
Certifier: check that all the vertices in V_T is in V (time complexity: $O(|V|)$); check that all edges in E_T is in E (time complexity: $O(|E|)$); check that all vertices in R is in V_T (time complexity: $O(|V|)$); check that $w(E_T) \leq B$ (time complexity: $O(|E|)$). Therefore, the time complexity is polynomial. □

4. Prove VERTEX COVER $\equiv_p$ CLIQUE

Proof. I prove this theorem by two steps.

(a) **INDEPENDENT SET $\equiv_p$ CLIQUE:** a vertex set S of a graph G is a independent set if and only if S is a clique in G', where G' is the complement of G.
"$\Longrightarrow$": Assume that $C(V_C, E_C)$ is a clique in $G'(V, E')$, where G' is the complement of $G(V, E)$. Assume that S is the independent set. If $\exists u \in V_C$ such that u is not in the independent set S, I have $\forall v \in \{V - V_C\}, e(u, v) \in E$ such that v is in the independent set S. The reason is that for each edge at most one of its endpoints can be allocated in the independent set S.
Set all of these vertices v form a vertex set $T \subseteq S$. Because for each edge at most one of its endpoints can be allocated in the independent set S, there should not be any edge between the vertices T in G. Therefore, G' should have all the edges between the vertices T as G' is the complement of G. It contradicts that $\forall v \subseteq T, v \in \{V - V_C\}$.
Therefore, if $C(V_C, E_C)$ is a clique in G', all the vertices in V_C is in the independent set S.
"$\Longleftarrow$": Assume that S is the independent set of G. $\forall u, v \in S$, there should not be an edge between u and v in G becasue for each edge at most one of its endpoints can be allocated in the independent set S. Therefore, there is an edge between u and v in G' becasue G' is the complement of G.
Therefore, the independent set S is a clique in G'.

(b) **VERTEX COVER $\equiv_p$ INDEPENDDENT SET.** This is proved in the slide.

(c) According to (a) and (b), VERTEX COVER $\equiv_p$ CLIQUE.

□