Lab08-Computational Complexity

CS214-Algorithm and Complexity, Xiaofeng Gao, Spring 2019.

$*$ If there is any problem, please contact TA Jiahao Fan or TA Mingran Peng.

$*$ Name:ChenxiaoYang   Student ID:517021910540   Email: cxyang1@outlook.com

1. Design a one-tape TM M that computes the function $f(x, y) = x - y$, where x and y are positive integers ($x > y$). The alphabet is $\{1, 0, \Box, \triangleright, \triangleleft\}$, and the inputs are x 1's, $\Box$ and y 1's. Below is the initial configuration for input $x = 7$ and $y = 3$. The result $z = f(x, y)$ should also be represented in the form of z 1's on the tape with the pattern of $\triangleright 111 \cdots 111 \triangleleft$.

Init:

	$\triangleright$	1	1	1	1	1	1	1	$\Box$	1	1	1	$\triangleleft$
	$\uparrow$ q_S												

(a) Please describe your design and then write the specifications of M in the form like $\langle q_S, \triangleright \rangle \to \langle q_1, \triangleright, R \rangle$. Explain the transition functions in detail.

(b) Please draw the state transition diagram using Microsoft Visio.

(c) Show briefly and clearly the whole process from initial to final configurations for input $x = 7$ and $y = 3$.

Solution.

(a)

Start State:

$$\langle q_S, \triangleright \rangle \to \langle q_1, \triangleright, R \rangle$$

Move Point to y:

$$\langle q_1, 1 \rangle \to \langle q_1, 1, R \rangle$$
$$\langle q_1, \Box \rangle \to \langle q_2, \Box, R \rangle$$

Subtract 1 in x and y:

$$\langle q_2, \Box \rangle \to \langle q_2, \Box, R \rangle$$
$$\langle q_2, 1 \rangle \to \langle q_3, \Box, L \rangle$$
$$\langle q_3, \Box \rangle \to \langle q_3, \Box, L \rangle$$
$$\langle q_3, 1 \rangle \to \langle q_2, \Box, R \rangle$$

End subtraction and Move Left:

$$\langle q_2, \triangleleft \rangle \to \langle q_4, \Box, L \rangle$$
$$\langle q_4, \Box \rangle \to \langle q_4, \Box, L \rangle$$

Put $\triangleleft$ in the result:

$$\langle q_4, 1 \rangle \to \langle q_5, 1, R \rangle$$
$$\langle q_5, \Box \rangle \to \langle q_H, \triangleleft, S \rangle$$

(b) The state transition diagram is drawn by draw.io

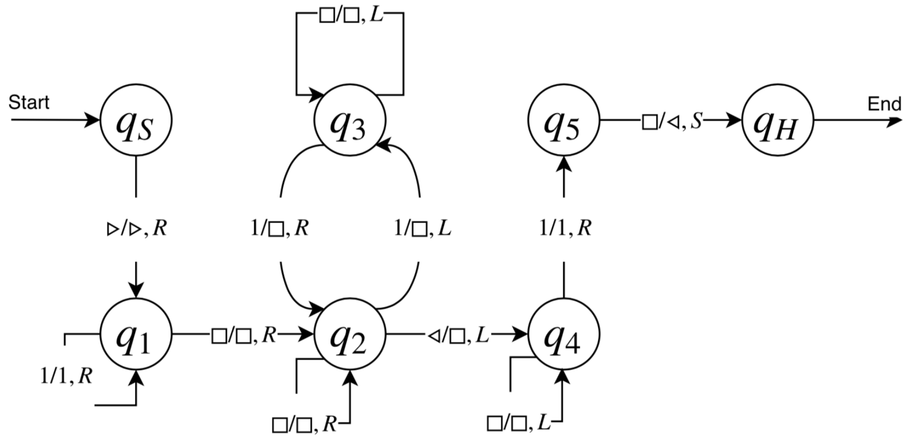

Figure 1: State transition diagram

(c) The whole process:

	$\triangleright$	1	1	1	1	1	1	1	$\square$	1	1	1	$\triangleleft$	
	$\uparrow$ q_S													

	$\triangleright$	1	1	1	1	1	1	1	$\square$	1	1	1	$\triangleleft$	
		$\uparrow$ q_1												

....

	$\triangleright$	1	1	1	1	1	1	1	$\square$	1	1	1	$\triangleleft$	
									$\uparrow$ q_1					

	$\triangleright$	1	1	1	1	1	1	1	$\square$	1	1	1	$\triangleleft$	
										$\uparrow$ q_2				

	$\triangleright$	1	1	1	1	1	1	1	$\square$	$\square$	1	1	$\triangleleft$	
									$\uparrow$ q_3					

	$\triangleright$	1	1	1	1	1	1	1	$\square$	$\square$	1	1	$\triangleleft$	
								$\uparrow$ q_3						

....

	$\triangleright$	1	1	1	1	1	1	$\square$	$\square$	$\square$	1	1	$\triangleleft$	
											$\uparrow$ q_2			

....

$\triangleright$	1	1	1	1	1	1	$\square$	$\square$	$\square$	$\square$	1	$\triangleleft$
						$\uparrow$						
						q_3						

....

$\triangleright$	1	1	1	1	1	$\square$	$\square$	$\square$	$\square$	$\square$	1	$\triangleleft$
											$\uparrow$	
											q_2	

....

$\triangleright$	1	1	1	1	1	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$	$\triangleleft$
					$\uparrow$							
					q_3							

....

$\triangleright$	1	1	1	1	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$	$\triangleleft$
												$\uparrow$
												q_2

$\triangleright$	1	1	1	1	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$
											$\uparrow$	
											q_4	

....

$\triangleright$	1	1	1	1	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$
				$\uparrow$								
				q_4								

$\triangleright$	1	1	1	1	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$
					$\uparrow$							
					q_5							

$\triangleright$	1	1	1	1	$\triangleleft$	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$	$\square$
					$\uparrow$							
					q_H							

$\square$

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

   (a) *PARTITION*: Given a finite set A and a size $s(a) \in \mathbb{Z}$ for each $a \in A$, is there a subset $A' \subseteq A$ such that $\sum_{a \in A'} s(a) = \sum_{a \in A - A'} s(a)$?

   (b) *CLIQUE*: Given a graph $G = (V, E)$ and a positive integer $K \leq |V|$, is there a subset $V' \subseteq V$ with $|V'| \geq K$ such that every two vertices in V' are joined by an edge in E ?

   (c) *ZERO-ONE INTEGER PROGRAMMING*: Given an integer $m \times n$ matrix A and an integer m-vector b, is there an integer n-vector x with elements in the set $\{0, 1\}$ such that $Ax \leq b$?

Solution.

(a) Certificate: a partition of set A
Certifier: compute the sum of each part and check if they are the same.

(b) Certificate: a subset of V
Certifier: check if the subset contains at least k elements and there is an edge between each pair of nodes in the subset.

(c) Certificate: an assignment of $\{0, 1\}$ to n-vector x
Certifier: compute Ax and check if each element of it is smaller than the corresponding element in b

$\square$

3. *SUBSET SUM*: Given a finite set A, a size $s(a) \in \mathbb{Z}$ for each $a \in A$ and an integer B, is there a subset $A' \subseteq A$ such that $\sum_{a \in A'} s(a) = B$?

   KNAPSACK: Given a finite set A, a size $s(a) \in \mathbb{Z}$ and a value $v(a) \in \mathbb{Z}$ for each $a \in A$ and integers B and K, is there a subset $A' \subseteq A$ such that $\sum_{a \in A'} s(a) \leq B$ and $\sum_{a \in A'} v(a) \geq K$?

   (a) Prove *PARTITION* $\leq_p$ *SUBSET SUM.*

   (b) Prove *SUBSET SUM* $\leq_p$ *KNAPSACK.*

   Solution.

   (a) Given a finite set A, a size $s(a) \in \mathbb{Z}$ for each $a \in A$. We compute the sum of all elements in A=A. If the sum is odd, then there is no such partition. Else if the sum is even, then the problem turns into finding if there is a subset $A' \subseteq A$ such that $\sum_{a \in A'} s(a) = 1/2 \sum_{a \in A} s(a)$.

   (b) Given a finite set A, a size $s(a) \in \mathbb{Z}$ for each $a \in A$ and an integer B. We give each element in A a value which is equal to its size. Also we let $K = B$. Then the problem turns into finding a subset $A' \subseteq A$ such that $\sum_{a \in A'} s(a) \leq B$ and $\sum_{a \in A'} v(a) \geq K$.

   $\square$

4. *3-SAT*: Given a set U of variables, a collection C of clauses over U such that each clause $c \in C$ has $|c| = 3$, is there a satisfying truth assignment for C?

   Prove *3-SAT* $\leq_p$ *CLIQUE.*

   Solution. Given an instance of 3-SAT, we construct an instance(G, k) that has a subset of size equal or larger than k.

   G contains 3 vertices for each clause, one for each literal. Initially, each pair of vertices is connected. Then, we remove the edge within each triangle to ensure that at most one vertex in the triangle is in a satisfying subset. Also, we remove all edges between x_k and $\bar{x_k}$ so that x_k and $\bar{x_k}$ won't both appear in the subset.

   If we can find a subset of size k in G, then we can assign true to each element in the subset so that ϕ is true. Also, if we can find a true assignment to ϕ, then there is a corresponding subset of size k in the construct graph.

   Let $\phi = (\bar{x}_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \bar{x}_2 \vee x_3) \vee \bar{x}_1 \vee x_2 \vee x_4)$, then the corresponding graph's complementary graph is shown in Figure 2. $\square$

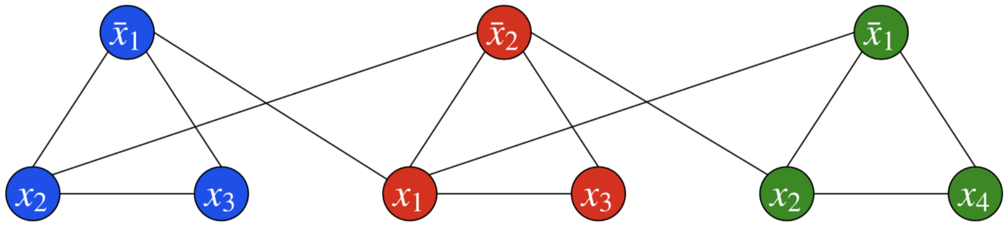

Figure 2: The complementary graph of the constructed G

5. Algorithm class is a democratic class. Denote class as a finite set S containing every students. Now students decided to raise a student union $S' \subseteq S$ with $|S'| \leq K$.

   As for the members of the union, there are many different opinions. An opinion is a set $S_o \subseteq S$. Note that number of opinions has nothing to do with number of students.

   The question is whether there exists such student union $S' \subseteq S$ with $|S'| \leq K$, that S' contains at least one element from each opinion. We call this problem *ELECTION* problem, prove that it is NP-complete.

   Solution. Given all opinions and a student union $S' \subseteq S$, we can check if S' contains at least one element from each opinion one by one. So we can certificate it in polynomial time and it's a NP problem.

   From the course we know that $3-SAT$ is NP-complete and $3-SAT \leq_p VERTEX-COVER$. Therefore, VERTEX-COVER problem is NP-complete.

   We claim that VERTEX-COVER problem can reduce to ELECTION problem. Given an arbitrary graph, we let each vertex be a student, and each edge be an opinion. Two corresponding vertices of the edge is the elements of the opinion.

   To check if there is a subset whose size is equal or smaller than k that covers all edges, we only have to check if there is a student union whose size is equal or smaller than k that contains at least one element from each opinion. Therefore, $VERTEX - COVER \leq_p ELECTION$. Since VERTEX-COVER problem is NP-complete, ELECTION problem is NP-complete. $\square$

Remark: You need to include your .pdf and .tex files in your uploaded .zip file.