NATIONAL UNIVERSITY OF SINGAPORE

CS 4232: Theory of Computation

Semester 1; AY 2019/2020; Midterm Test 2

Time Allowed: 40 Minutes

INSTRUCTIONS TO CANDIDATES

1. Please write your Student Number. Do not write your name.
2. This assessment paper consists of FOUR (4) questions and comprises NINE (9) printed pages.
3. Students are required to answer **ALL** questions.
4. Students should answer the questions in the space provided.
5. This is a **CLOSED BOOK** assessment.
6. Every question is worth FIVE (5) marks. The maximum possible marks are 20.

STUDENT NO: ______________________________

This portion is for examiner's use only

Question	Marks	Remarks
Question 1:		
Question 2:		
Question 3:		
Question 4:		
Total:		

Let $\Sigma = \{0, 1, 2, 3\}$. Recall that a generalised homomorphism is a mapping from regular sets to regular sets such that $h(\emptyset) = \emptyset$, $h(\{\varepsilon\}) = \{\varepsilon\}$, $h(L \cup H) = h(L) \cup h(H)$, $h(L \cdot H) = h(L) \cdot h(H)$ and $h(L^*) = (h(L))^*$. List all h satisfying the following conditions:

1. h is a generalised homormophism;
2. $h(L) = \bigcup_{w \in L} h(\{w\})$ for all regular sets L;
3. $h(\{0, 1, 2, 3\}) = \{4, 44, 444\}$;
4. $h(\{0123\}) = \{44444, 444444\}$;
5. $h(\{2233\}) = \{4444\}$.

Here each generalised homormorphism satisfying the above conditions can be listed by determining what $h(\{a\})$ is for all $a \in \{0, 1, 2, 3\}$.

Solution. By the third condition, every $h(\{a\})$ is a subset of $\{4, 44, 444\}$; by the fourth condition, each set $h(\{a\})$ is not empty. By $h(\{2233\}) = \{4444\}$, $h(\{2\}) = \{4\}$ and $h(\{3\}) = \{4\}$. It can not be that both $h(\{0\})$ and $h(\{1\})$ contain 4 as then $4444 \in h(\{0123\})$, what is not the case. Furthermore, exactly one of $h(\{0\})$ and $h(\{1\})$ has two elements, the other one has one element. One of $h(\{0\})$ and $h(\{1\})$ must have the element 444. So one has the following choices:

1. $h(\{0\}) = \{4\}$, $h(\{1\}) = \{44, 444\}$, $h(\{2\}) = \{4\}$ and $h(\{3\}) = \{4\}$;
2. $h(\{0\}) = \{44, 444\}$, $h(\{1\}) = \{4\}$, $h(\{2\}) = \{4\}$ and $h(\{3\}) = \{4\}$.

So there are in total two such h.

Let $L = \{0^n10^n : n \geq 1\}$ and $H = L^+$ where L^+ is the Kleene plus of L. Provide a grammar in Greibach Normal Form for H and give a sample derivation for 01000100.

As H does not contain ε, a grammar for H is in Greibach Normal Form if every rule is of the form $A \to bw$ where A is a nonterminal, b a terminal and w a possibly empty word of nonterminals.

Solution. The grammar is $(\{0,1\}, \{S,T,U\}, P, S)$ where the rules in P are the following: $S \to 0TUS \,|\, 0TU$, $T \to 0TU \,|\, 1$, $U \to 0$.

The sample derivation is $S \Rightarrow 0TUS \Rightarrow 01US \Rightarrow 010S \Rightarrow 0100TU \Rightarrow 01000TUU \Rightarrow 010001UU \Rightarrow 0100010U \Rightarrow 01000100$.

Consider the grammar $(\{0\}, \{S, T\}, \{S \to ST|TS|TT|0, T \to ST|TS|TT|0\}, S)$. Determine the number of derivation trees of 0000 in this grammar.

Solution. Note that both S and T have the same rules; so the entries for S and T are the same. Let $F(A, w)$ be the number of derivation trees which allow to derive w from the symbol A for $A = S, T$. The following can be derived from the Algorithm of Cocke, Kasami and Younger (all entries on each level of the pyramid would be the same):

- $F(S, 0) = 1$, $F(T, 0) = 1$;
- $F(S, 00) = F(S, 0) \cdot F(T, 0) + F(T, 0) \cdot F(S, 0) + F(T, 0) \cdot F(T, 0) = 3$, $F(T, 00) = 3$; note that using $F(S, w) = F(T, w)$ in the grammar, one can just state this as $F(S, 00) = 3 \cdot F(S, 0) \cdot F(S, 0)$;
- $F(S, 000) = 3 \cdot F(S, 0) \cdot F(S, 00) + 3 \cdot F(S, 00) \cdot F(S, 0) = 18$ and $F(T, 000) = 18$;
- $F(S, 0000) = 3 \cdot F(S, 0) \cdot F(S, 000) + 3 \cdot F(S, 00) \cdot F(S, 00) + 3 \cdot F(S, 000) \cdot F(S, 0) = 54 + 27 + 54 = 135$.

As the start symbol is S, the overall number of derivation trees is 135.

Recall that the sequence of Fibonacci numbers is defined by

Fibonacci(0) = 0, Fibonacci(1) = 1 and
Fibonacci($n+2$) = Fibonacci(n) + Fibonacci($n+1$).

Write a function Nextfibonacci with

$$\text{Nextfibonacci}(x) = \min\{y \geq x : \exists z \in \mathbb{N}\,[y = \text{Fibonacci}(z)]\}$$

which finds for input x the smallest Fibonacci number y with $y \geq x$.

Register machine programs can use conditional and unconditional goto-commands, compare ($<, \leq, =, \geq, >, \neq$) and add (+) and subtract (−) registers and natural numbers and has the input x in the register R_1. The return command provides the output of the function; for example, Return(R_7) returns the content of register R_7 as output of the function. Write the function as one program without using any macros.

Solution. The program can be made as follows.

Line 1: Function Nextfibonacci(R_1);
Line 2: $R_2 = 0$; $R_3 = 1$;
Line 3: $R_4 = R_2 + R_3$;
Line 4: If $R_1 \leq R_2$ Then Goto Line 7;
Line 5: $R_2 = R_3$; $R_3 = R_4$;
Line 6: Goto Line 3;
Line 7: Return(R_2).