NATIONAL UNIVERSITY OF SINGAPORE

CS 3231 – Theory of Computation

Semester 1; AY 2021/2022; Midterm Test

Time Allowed: 60 Minutes

INSTRUCTIONS TO CANDIDATES

1. Please write your Student Number. Do not write your name.
2. This assessment paper consists of SIX (6) questions and comprises THIRTEEN (13) printed pages.
3. Students are required to answer **ALL** questions.
4. Students should either answer the questions in the space provided on the pdf of the question paper or hand-write on A4 paper, with each question on an extra page having question number and student number on top and questions being ordered before scanning.
5. This is an **CLOSED BOOK** assessment **WITH HELPSHEET**.
6. You are not permitted to communicate with other people except inviligators during the exam and you are not allowed to post in forums or other such entries during the exam and you are not allowed to read the internet, the Luminus or other sources except the question paper.
7. Every question is worth FOUR (4) or FIVE (5) marks. The maximum possible marks are 26.
8. The file should be submitted to LUMINUS FILES and should have the name STUDENTNUMBER-MT-CS3231.pdf, say A01234567X-MT-CS3231.pdf

STUDENT NO: ______________________________

Let $\Sigma = \{0, 1, 2\}$. Construct a regular expression which contains all strings of ternary numbers which are not multiples of six (6). The numbers should not have leading zeroes and the first digit should therefore be 1 or 2. Explain how the expression was formed.

Here the regular expression can use finite sets given as a list of words in it within set brackets, Kleene star, Kleene plus, union, concatenation and intersection. Kleene plus and Kleene star bind more than all binary operations and concatenation binds more than intersection and intersection binds more than union. In doubt, use brackets to make the expression clear.

Solution. Numbers which are divisible by 3 end in ternary numbers with a 0. So such numbers should not be even. Odd numbers have an odd number of 1 in their representation. This can be achieved by starting the number with either a 1 or a prefix of the form 2 followed by $0, 2$ arbitrarily often and then followed by 1. After that one must make sure that all 1s occur an even time, thus the expression is the Kleene star over a union of $\{0, 2\}$ and the set $\{1\} \cdot \{0, 2\}^* \cdot \{1\}$. After that the number is followed by a 0 and ends. If the number ends with 1 or 2, the only requirement is that it also starts with one of these digits, as the number then is not a multiple of 3 and thus also not a multiple of 6. The union of these two expressions gives the full expression. Furthermore, one has to add the one-digit numbers $\{1, 2\}$ to the set. Thus

$$(\{2\}\cdot\{0\}^*)^*\cdot\{1\}\cdot(\{0,2\}\cup(\{1\}\cdot\{0,2\}^*\cdot\{1\}))^*\cdot\{0\} \cup \{1,2\}\cdot\{0,1,2\}^*\cdot\{1,2\}$$
$$\cup \{1,2\}$$

is a regular expression generating the required set. Here concatenation binds stricter than union. Note that regular expressions are not unique and that there are many solutions.

Consider the language L of all binary words which are nonempty and which are not palindromes but for which there is one symbol which one can flip (0 to 1 or 1 to 0) such that ther resulting word is a palindrome. Construct a grammar (N, Σ, P, S) of L which is of one of the following forms:

1. Regular: Every rule in P is of the form $A \to v$ or $A \to wB$ where $v, w \in \Sigma^*$ and $A, B \in N$;
2. Linear: Every rule in P is of the form $A \to vBw$ or $A \to v$ with $v, w \in \Sigma^*$ and $A, B \in N$;
3. Context-free: Every rule in P is of the form $A \to w$ where $A \in N$ and $w \in (N \cup \Sigma)^*$;
4. Context-sensitive: Every rule in P is of the form $l \to r$ where l contains at least one non-terminal and $|l| \leq |r|$.

Construct the grammar according to the first of the four option which is possible and explain why none of the options before it is possible.

Here a palindrome is a word w with $w = w^{mi}$, that is, the mirror image of w is equal to w.

Solution. The best possible choice is linear. The grammar is the same as that for palindromes, but one has to make sure that exactly one error is built into the binary word. Therefore one needs two non-terminals: S when the error is not yet built in and T when the error is built in. The transition from S to T makes one nonsymmetric step. S and T are always in the middle of the word. Thus the grammar $(\{S, T\}, \{0, 1\}, P, S)$ has the following rules in the set P: $S \to 0S0|1S1|01|10|0T1|1T0$, $T \to 0T0|1T1|00|11|0|1$. One can see as follows that the language is not regular.

If L would be regular then also $L \cap 00^*10^*1$ would be regular. The outmost pair $1, 0$ produces an error, therefore the inner part must be a palindrome with a 1 in the middle. So the derivatives L_{0^n} with $n > 0$ have the shortest word $10^{n-1}1$ in the language, all further words are of the form $0^m10^{m+n-1}1$ with $m > 0$. Thus the derivatives of L_{0^n} and L_{0^m} with $0 < n < m$ are distinct and there are infinitely many derivatives. By the Theorem of Myhill and Nerode, the language is not regular. Thus one cannot use a regular grammar and has to construct a linear grammar what was shown above to be possible.

Construct a complete dfa which recognises all ternary words where there are exactly two ones and one two. Such words include 112, 10201, 21100, 00000100002000100. A complete dfa has a transition function δ which assigns to each pair $(\mathit{state}, \mathit{symbol})$ exactly one new state.

Solution. The states of the dfa are of the form (a, b) where a is the number of 1s seen so far and b is the number of 2s. Furthermore, when $a = 3$, the dfa stops counting 1s, when $b = 2$, it stops counting 2s. So the transition function δ is as follows: $\delta((a, b), 0) = (a, b)$, $\delta((a, b), 1) = (\min\{a + 1, 3\}, b)$, $\delta((a, b), 2) = (a, \min\{b + 1, 2\})$. $(0, 0)$ is the start state and $(2, 1)$ is the unique accepting state.

One can reduce the number of states by merging all states where $a = 3$ or $b = 2$ into one single always rejecting state. Thus the dfa can be constructed with 7 states: $(0, 0), (1, 0), (2, 0), (0, 1), (1, 1), (2, 1), r$ where r is the merger of $(3, 0), (3, 1), (3, 2)$, $(2, 2), (1, 2), (0, 2)$.

Let L be the set of all words over $w \in \{0,1,2\}$ such that either w has a subword of the form uu of length at least 6 (that is, $w \in \{0,1,2\}^* \cdot \{uu\} \cdot \{0,1,2\}^*$ with $|u| \geq 3$) or there is an odd prime number which divides the length $|w|$ of w.

Furthermore, consider the following variant of the block pumping lemma: The language L satisfies this pumping lemma with constant k iff for all words of the form $u_0 u_1 \ldots u_k \in L$ where all blocks are nonempty, there are i, j with $1 \leq i \leq j \leq k-1$ such that, for all $h > 0$, $u_0 \ldots u_{i-1}(u_i \ldots u_j)^h u_{j+1} \ldots u_k \in L$.

Find the smallest constant k such that this variant of the block pumping lemma is satisfied or prove that such a constant does not exist and that L does not satisfy this variant of the block pumping lemma.

Furthermore, either construct an nfa for the language L or show that no nfa recognises the language L. An nfa needs to have one start state and reads in every transition exactly one symbol from the word to be processed, it does not have ε-transitions. An nfa accepts a word w iff there is a run of the nfa on the word such that after processing the complete word the run ends with an accepting state.

Solution. The language L satisfies the variant of the block pumping lemma with constant $k = 4$. If $k = 3$, one could consider the word 012102. This word is square-free, but it is in L due to its length having the prime-factor 3. Now consider the splitting of this word into u_0, u_1, u_2, u_3 as $012, 1, 0, 2$. The following pumps have to be considered: 1, 0 and 10. If one repeats the one-digit pumps at least 5 times, they contain a square of length 6 and are in L; if one repeats the two-digit pump 10 at least three more times, it has the subword 10101010 which is a square of length 8. Thus only shorter pumps might go out of L and this only happens if the length is 8 which is a number not having an odd prime factor. The following pumped words have the length 8: 01211102, 01210002, 01210102. All of these do not contain a square of length 6 or length 8 and are thus not in L, hence the pumping lemma is not satisfied for $k = 3$. For $k = 4$, any word which can be split into nonempty blocks u_0, u_1, u_2, u_3, u_4 satisfies that the pump $u_1 u_2 u_3$ has at least length 3 and when it is repeated at least once, the square $uu = u_1 u_2 u_3 u_1 u_2 u_3$ of length at least six is contained as a subword in the pumped up word, hence the variant of the block pumping lemma considered is satisfied with constant $k = 4$.

If the language L would be recognised by an nfa, then it would be regular and thus its complement H would have to satisfy the tradiational pumping lemma (and also the other variants). The complement H contains all square-free ternary words whose length is a power of 2, as powers of two have no odd prime factors (plus some other words). However, if H is regular then any sufficiently long word $w \in H$ satisfies that it can be split into x, y, z with $y \neq \varepsilon$ such that $xy^6 z \in H$. However, y^6 is a square of length at least six and thus, by definition, $w \in L$. It follows that $w \notin H$ and H does not satisfy the traditional pumping lemma and therefore neither H nor L are regular, in particular L is not recognised by any nfa.

Quninary numbers are numbers based on the digits $0, 1, 2, 3, 4$ and sample numbers are 13 (eight) and 111 (thirty-one). Recall that a homomorphism h is a mapping from words to words such that $h(v \cdot w) = h(v) \cdot h(w)$ for all words and the concatenation operation $\cdot$; furthermore, a homomorphism satisfies $h(\varepsilon) = \varepsilon$ and is given by the values of $h(a)$ for all letters a. $h(a)$ can be any word and does not need to have length 1.

Either construct a homomorphism with the below properties or say that such a homomorphism does not exist. The homomorphism should map decimal numbers to quinary numbers such that the following conditions (a)–(c) hold:

(a) the homomorphism is one-one,

(b) a number without leading zeroes is mapped to a number without leading zeroes,

(c) a number x in decimal is a multiple of 3 if and only if its image $h(x)$ is a multiple of 3 as a quinary number.

It does not matter what the homomorphism does with numbers having leading zeroes.

If h is constructed, explain how h works; if it is stated that there is no such h, then say why h does not exist.

Solution. The homomorphism is given by the rules $h(0) = 11$, $h(1) = 12$, $h(2) = 13$, $h(3) = 14$, $h(4) = 20$, $h(5) = 21$, $h(6) = 22$, $h(7) = 23$, $h(8) = 24$, $h(9) = 30$. Note that this solution is not unique and other homomorphisms h are also possible. All these images of digits are two-digit quinary numbers and they satisfy a and $h(a)$ have modulo three in their respective numerical systems the same value 0, 1 or 2, respectively; here $h(a)$ has the value $a + 6$ written as a quinary number. As the quinary blocks have all length 2, they can be viewed as digits in a system with base 25. This base is one larger than a multiple of 3. Thus both the decimal system (base 10) and the system base 25 satisfy that a number is a multiple of 3 iff their digits (modulo 3) sum up to a multiple of 3. Thus the homomorphism h preserves the remainder by 3, as each decimal digit is translated into a block with the same remainder in the base 25 system.

Consider the language $\{0^n 1^m 0^{n+m+1} : n, m \geq 1\}$ and provide for this language grammars in (a) Chomsky Normal Form and (b) Greibach Normal Form. One of the grammars should have at most three and the other one at most six non-terminals (the student can choose which has only three). Explain the grammars.

For this, recall that for languages without the empty word, the Chomsky Normal Form only contains rules of the form $A \to a$ and $A \to BC$ and the Greibach Normal Form contains only rules of the form $A \to aw$ where A, B, C are non-terminals, a is a terminal symbol and w is a possibly empty word of non-terminals.

Solution. A normal context-free grammar for this is given by $S \to 0S0|0T0, T \to 1T0|100$. Each right side of the rule contains either a non-terminal and two terminals or three terminals. An application of a rule and S stands for building up the n symbol pairs $0,0$ and rules with T stand for builindg up the m symbol pairs $1,0$ in the word $0^n 1^m 0^{m+n+1}$. The condition that $n \geq 1$ is maintained by having at least the state $S \to 0T0$ in any terminating derivation and the condition that $m \geq 1$ plus the extra trailing 0 are maintained by the rule $T \to 100$ being the last rule which must be applied to derive an all-terminal word.

The grammar in Chomsky Normal Form has the non-terminals S, T, V, W, X, Y, the terminals $0, 1$, the start symbol S and the rules $S \to VX$, $X \to SV|TV$, $T \to WY$, $Y \to TV|VV$, $V \to 0$, $W \to 1$. The idea is here that V, W are place-holders for the non-terminals $0, 1$ and X, Y are used to break three-letter right sides into two-letter right sides by being used as intermediate non-terminals when producing the corresponding right side.

The Greibach Normal Form needs only one place-holder for the trailing 0 and besides that is identical with the given context-free grammar. So it has the non-terminals S, T, U, the terminals $0, 1$, the start symbol S and the rules $S \to 0SU|0TU$, $T \to 1TU|1UU$, $U \to 0$. In other words, in the above context-free grammar, the trailing 0s are replaced by Us on the right sides and the rule $U \to 0$ is added in.

END OF QUESTION PAPER.