RMIT UNIVERSITY

Computing Theory
COSC 1107/1105
Final Exercise

Assessment Type	**Individual assignment.** Submit online via Canvas → Assignments → End of semester exercise. Marks awarded for meeting requirements as closely as possible. Clarifications/updates may be made via announcements/relevant discussion forums.
Due Date	Saturday 6th November 2021, 9.00am
Marks	110 worth 50% of your final assessment

1 Overview

This assessment requires you to demonstrate your knowledge of the key concepts in the course.

Please note that this is an individual assessment; you are expected not to communicate with any other person in any way about this exercise for the entire duration of it.

2 Assessment details

1. Consider the grammar derivations below.

   $S \Rightarrow 00S1 \Rightarrow 0000S11 \Rightarrow 00000S111 \Rightarrow 00000\#A$ $1 \Rightarrow 00000\#C\#111 \Rightarrow 00000\#cDd\#111 \Rightarrow 00000\#c2d\#111$

   $S \Rightarrow \#A\# \Rightarrow \#aAb\# \Rightarrow \#aaBbb\# \Rightarrow \#aacCdbb\# \Rightarrow \#aaccCddbb\# \Rightarrow \#aacccDdddbb\# \Rightarrow \#aaccc3dddbb\#$

   $S \Rightarrow 0S1 \Rightarrow 0\#A\#1 \Rightarrow 0\#aaAbb\#1 \Rightarrow 0\#aaBbb\#1 \Rightarrow 0\#aaCbb\#1 \Rightarrow 0\#aacCddbb\#1 \Rightarrow 0\#aaccDdddbb\#1 \Rightarrow 0\#aacc2dddbb\#1$

   (a) From the above derivations, construct rules that must exist in any context-free grammar G for which these derivations are correct. (2 marks)

   (b) Assuming that these are all the rules in G, give $L(G)$ in set notation. (3 marks)

   (c) Is the grammar G context-sensitive? Explain your answer. (2 marks)

   (d) What happens to $L(G)$ if the rule $S \rightarrow SS$ is added to G? Explain your answer (but there is no need to give the new language in set notation). (3 marks)

   (e) Construct a context-free grammar for the language L_2 below. (4 marks)
   $L_2 = \{a^i\ 0^j\ w\ c^l\ e\ d^l\ w^R\ 1^k\ b^{2i} \mid w \in \{2,3\}^*, k \geq j, i, j, k \geq 0, l \geq 1\}$

   (f) Explain why there is not necessarily a deterministic pushdown automaton for L_2. (2 marks)

   (2+3+2+3+4+2 = 16 marks)

2. The following discussion was discovered in some ancient archives. There are between 6 and 12 incorrect statements in the paragraph below. Identify all incorrect statements and justify each of your answers.

   Note: A single sentence counts as one statement.

   ”The Chomsky Hierarchy is a way of classifying languages and the classes of grammars that generate them. The distinct classes of grammar are labelled as Type 0 to Type 4 inclusive. Each class of grammar is distinct, meaning that no grammar appears in more than one class. For each class of grammar there is a corresponding class of automata, and these automata can be either deterministic or non-deterministic. For every nondeterministic automaton in any of these classes there is an equivalent deterministic automaton which recognises the same language. It is also the case that for every possible input for every automaton in these classes, the computation on that input always terminates, although this may take a very long time. This means that we can divide the decidable problems into two classes – the tractable ones, which can be decided in at most polynomial time on a deterministic Turing machine, and the intractable ones which take at least exponential time on a deterministic Turing machine. Some well-known examples of intractable problems include the vertex cover problem, the Hamiltonian circuit problem, the Travelling Salesperson problem and the Halting problem for Turing machines. It is remarkable that many of these intractable problems are related by being NP-complete, which means they are all of a similar level of compl
   polynomial space on a nond s
   be in NP, but also be at least as har
   by a process known as reduct
   that it can be reduced to some already known NP-complete proble
   from the early 1970’s, which showed that 3SAT is NP-complete, a l
   problems and their relationships has been built up over the years lems
   include the knapsack problem, the bin packing problem, integer factorisation and the 3-colouring problem. It is strongly suspected that all NP-complete problems are intractable, i.e. cannot be solved in polynomial time. In order to show that P $\neq$ NP, it would be sufficient to show that one NP-complete problem is certainly intractable. NP-complete problems also have the intriguing property that a purported solution can be checked in polynomial time, which makes them potentially useful for zero-knowledge proofs. ” (12 marks)

3. (a) Prove that Empty Language problem for Turing machines is undecidable by reducing the Halting problem to it. Make sure all the details of the reduction are clearly specified. You may use a diagram to assist with this if you wish. (4 marks)

   (b) A similar problem is the All Inputs problem for Turing machines. How similar is the proof for the undecidability of this problem to the above proof for the Empty Language problem? Explain your answer. (4 marks)

   (c) The sun is around 4.5 billion years old, and is expected to use up its hydrogen reserves (leading to it becoming a red giant) in about 5 billion years. The Tower of Hanoi is a game that is known to require $2^n - 1$ moves to complete for a tower of height n. The 4-player Platypus game is known to require $n(n+1)(n+2)(n+3)/24$ matches for a tournament with n entries. Assuming that moves in the Tower of Hanoi take 0.1 seconds each and $n = 64$, and 4-player Platypus matches take 0.001 seconds each and $n = 500,000$, calculate which process is more complete by the time the sun uses up its hydrogen reserves. In other words, at that time, which will be greater – the percentage of the Tower of Hanoi moves that are completed out of the total when $n = 64$ or the percentage of the matches for a tournament of n machines that are completed out of the total when $n = 500,000$? Explain your answer and include all relevant calculations. (4 marks)

(d) How does your answer change if the moves for the Tower of Hanoi now take 0.01 seconds and those for the Platypus game now take 0.01 seconds? Explain your answer. (2 marks)

(4+4+4+2 = 14 marks)

4. "Everything in triplicate" is the motto of the University of Warthogs, which means that student numbers are of the form www where $|w| = 3$ and $w \in \{0, 1, 2, 4, 5, 7, 8\}^*$. The digits 3, 6 and 9 are not used in such numbers, as these are considered sacred digits. For example 014014014, 778778778 and 224224224 are valid student numbers.

(a) Construct a Turing machine to recognise such numbers. Note that any string of length 8 or less should be rejected, as should any string of length 10 or more. In addition, any string of length 9 containing a 3, 6 or 9 should be rejected. (3 marks)

(b) Which of the following machines could also be used to recognise such numbers? Briefly justify each of your answers. (2 marks)

- A non-deterministic PDA
- A deterministic PDA
- A non-deter
- A determini
- A linear-bou

(c) The Chancellor of the University of Warthogs, Pumbaa the M at limiting the number of students in his university to a few hund he then decrees that student numbers can now be of the form or ww^Rw or www^R or ww^Rw^R where $|w| = 3$ and $w \in \{0, 1, 2, 4, 5, 7, 8\}^*$

Which of the following machines can be used to recognise the new student numbers? Briefly justify each of your answers. You do not have to explicitly construct any machines in your answer. (3 marks)

- A deterministic Turing machine
- A non-deterministic PDA
- A non-deterministic finite state automaton (NFA)
- A deterministic finite state automaton (DFA)

(d) Having finally received some better advice, Pumbaa further decrees that from now on, all student numbers will be of the form ww^Rw where $|w| \geq 3$ and $w \in \{0, 1, 2, 4, 5, 7, 8\}^*$.

Given these changes, which of the following machines can be used to recognise this latest definition of valid student numbers? Briefly justify each of your answers. You do not have to explicitly construct any machines in your answer. (3 marks)

- A deterministic Turing machine
- A non-deterministic PDA
- A non-deterministic finite state automaton (NFA)
- A deterministic finite state automaton (DFA)

(e) After a particularly bad nightmare, Pumbaa (who changes his mind like the wind) now insists that in addition to the conditions in the previous question, all numbers must have the property that the w in ww^Rw cannot repeat any digits. In other words, the only valid student numbers are ww^Rw where $|w| \geq 3$ and $w \in \{0, 1, 2, 4, 5, 7, 8\}^*$ and there are no repeated digits in w. For example, 122442211224 is no longer a valid student number as the string 1224 has a repeated 2. Does this change your answer to the previous question? If so, why and how? If not, why not? (3 marks)

(3+2+3+3+3 = 14 marks)

5. (a) Construct a Turing machine M_1 which given a string of length 7 over $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ performs as follows. (3 marks)

- The machine only halts in an accepting state if the input string has exactly 7 digits and commences with 3. Strings of any other length are rejected. Strings of length 7 that do not commence with 3 cause the machine to loop forever.
- If the last digit is 1, 3, 7 or 9, the machine replaces each of these digits with 5, and each of the digits 2, 4, 6 and 8 with 0.
- If the last digit is 2, 4, 6 or 8, the machine replaces each of these digits with 5, and each of the digits 1, 3, 7 and 9 with 0.
- If the last digit is 0 or 5, the machine loops forever.

(b) Construct a Turing m ,3, 4, 5} performs as follows. (3 marks)

- If there are no 5's i
- Otherwise
  - if there are an even number of 5's in the input string, th with a 2.
  - If there are an odd number of 5's in the string, the machine replaces each 5 with a 1.
- All 0's in the string are replaced with a 3.

(c) Consider the machine formed by combining M_1 and M_2 as above in sequence into a new machine M_3, so that M_3 first performs as M_1 does, rewinds the tape head to the appropriate point on the tape and then performs as M_2 does. Clearly the behaviour of M_3 will depend on the behaviour of M_1 and M_2. (4 marks)

- For which inputs does M_3 halt with rejection?
- For which inputs does M_3 halt with acceptance?
- For which inputs does M_3 not halt?

In each case, justify your answer.

(d) Consider now the machine M_0, which is the same as M_1 except that rather than deterministically replacing 1,3,7, and 9 with 5, M_0 *nondeterministically* replaces each of these digits with one of 5 or 2, and rather than deterministically replacing 2,3,6, and 8 with 5, M_0 *nondeterministically* replaces each of these digits with one of 5 or 3. For example, if there is a transition such as $\delta(q_i, 3) = (q_j, 5, R)$ in M_1, there are 2 corresponding transitions in M_0, i.e. $\delta(q_i, 3) = \{(q_j, 5, R), (q_j, 2, R)\}$ in M_0. (4 marks)

The same combination technique as above is used to combine M_0 and M_2 into M_4.

- For which inputs does M_4 halt with rejection?
- For which inputs does M_4 halt with acceptance?

- For which inputs does M_4 not halt?

In each case, justify your answer.

(3+3+4+4 = 14 marks)

6. (a) Prove that the language $L_1 = \{ww \mid w \in \{0,1,2\}^*\}$ is not regular by using the Pumping Lemma. Use the string 1^n2^n at an appropriate point in the proof. (5 marks)

(b) Write out the proof of the same result which uses the string 2^n10 and $i = 3$ instead. Which steps are different? Which steps remain the same? (4 marks)

(c) There are some errors in the proof below that the language $L_2 = \{1^i\ 2^j\ 3^i\ 4^i \mid i,j \geq 0\}$ is not context-free. Find and correct all such errors. The best way to do this is to write out the correct proof, highlighting the differences between the correct proof and the one below. (4 marks)

Claim: The language $L_2 = \{1^i\ 2^j\ 3^i\ 4^i \mid i,j \geq 0\}$ is not context-free.

Proof: Assume L_2 is regular. Then the Pumping Lemma applies and so for all $n \geq 1$ such that for some $w \in L$ such that $|w| \geq n$, $w = xyzuv$ where

- $|yzu| \leq n$
- $y \neq \lambda$ and $u \neq \lambda$
- $xy^izu^iv \in L$

Choose $w = 1^n2$ mma $w = xyzuv =$ $1^n2^n3^n4^n$ and $|y$ st two different digits.

Now if y or u contains 12, or 23 or 34, then x e some digits out of order.

So both y and u contain at most one type of digit. But thi $xy^2zu^2v \notin L$, as then there will not be equal numbers of 1's, 3's and 4's in $xyzu^3v$.

This is a contradiction, and so L_2 is not context-free.

(d) Give a PDA for the language $L_3 = \{1^i\ 2^j\ 3^j\ 4^i \mid i,j \geq 0\}$. Is your machine deterministic or non-deterministic? Briefly explain your answer. (2 marks)

(e) Is there a PDA for the language $L_4 = \{w \mid w \in \{1,2,3,4\}^*, n_1(w) = n_4(w), n_2(w) = n_3(w)\}$? Note that this is similar to L_3 but without any constraint on the order of the digits (i.e. the number of 1's and 4's in w must be the same, and the number of 2's and 3's in w must be the same). Explain your answer. Note that there is no need to either construct a PDA or give a proof that there is none to answer this question; a few sentences of argument will suffice. (3 marks)

(f) Give a context-free grammar for $L_5 = \{2\#\ 1^i\ 2^j\ 0\ 3^j\ 4^i\#3 \mid i,j \geq 0\}$ with at most 5 rules. The number of rules is the number of different right-hand sides in the grammar; for example, the grammar below has 6 rules. (2 marks)

$S \rightarrow aA|Aa|a$
$A \rightarrow aBaa|Baa$
$B \rightarrow abc$

(5+4+4+2+3+2 = 20 marks)

7. Gregor the Grammar Goblin hears about your work in Computing Theory, and is of course very impressed. He particularly loves the idea of grammars generating languages, and that in principle, everyone could specify their own grammar, and hence their own language. He is also an

entrepreneur, and during a conversation with you about the Chomksy hierarchy, he has an idea for a new business. This is for customers to send him an unrestricted grammar, from which he will then generate all words in the language (up to a limit of 1,000), which he will then arrange via a new artistic algorithm of his in a visually appealing manner. He will then charge the customer a very large sum of money for the artistic output. However, Gregor is not a fan of swear words, and so he wants to incorporate in the process a means to check the words generated by the grammar against his list of offensive words. If any of the words on Gregor's list are in the language specified by the customer's grammar, Gregor will refuse to generate the artistic output, and instead fine the customer for offensive language. Naturally Gregor looks to you to develop this process, and insists that this process must work for any possible grammar that the customer may have.

(a) Explain why Gregor's idea will not work, no matter how much computing power he may use. (4 marks)

(b) Gregor also hears on his social media channels about universal Turing machines, thinks they are very cool, and suggests to you that a solution to his original problem may be to encode the customer's unrestricted grammar as an input to a universal Turing machine. Explain why this approach will be no help. (2 marks)

(c) Suggest a way in which Gregor can accomplish something like his original intention by an appropriate relaxation of his policy. (3 marks)

(d) Gregor is rather stubb [illegible] ess of checking for swear [illegible] erarchy, if any, could be used for this purpose? Explain your answer. (3 marks)

(4+2+3+3 = 12 marks)

8. Consider the definitions below.

- A *universal Turing machine* U is a Turing machine that takes the encoding of a Turing machine M and an input string w and emulates the computation of M on w. More specifically, U takes as input *code(M)code(w)*, (where *code()* is a function which encodes Turing machines and inputs into the input language of U) and for every configuration in the computation of M on w, there is a corresponding configuration in the computation of U on *code(M)code(w)*.
- A universal Turing machine U is **steadfast** if for every M and w, whenever M terminates on w, U terminates on *code(M)code(w)*.
- A universal Turing machine U is **dedicated** if for every M and w, whenever M does not terminate on w, U does not terminate on *code(M)code(w)*.
- A universal Turing machine is **constant** if U is steadfast **and** dedicated.

(a) Is it possible to have a universal Turing machine which is steadfast but not dedicated? Is it possible to have a universal Turing machine which is dedicated but not steadfast? Explain your answers. (4 marks)

(b) Should it be a requirement for a universal Turing machine to be constant? Or is it reasonable for a universal Turing machine not to be constant? Explain your answer, using appropriate examples. (4 marks)

(4+4 = 8 marks)

3 Submission

You should submit one PDF file containing your answers to the questions. Do not use a .zip file. **No file formats other than PDF will be accepted.**