Curtin University

Venue ____________________

Student Number |__|__|__|__|__|__|__|__|

Family Name _____________________

First Name _____________________

Department of Computing

EXAMINATION

Semester 2 & Trimester 3B, 2015

COMP3002 Theoretical Foundations of Computer Science

This paper is for Bentley Campus students.

This examination has a total of 120 marks.

Examination Duration: 120 minutes

Reading Time: 10 minutes

Exam Conditions:

This is a CLOSED BOOK exam - no text books or written materials permitted

Students are permitted to write notes during reading time in the margins of the exam paper

This paper can be released to students after the exam

The students exam paper can be released to students after the exam

No calculators are permitted in this exam

Materials Permitted In The Exam Venue:

none

Materials To Be Supplied To Students:

1 x 16 page answer book

Instructions To Students:

For Examiner Use Only

Q	Mark
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	

Total ________

Section 1 - Classification.

Problems 1 to 6 in this section each describe a problem in English, set notation or in terms of strings. It is your task to do the following, where applicable, for each of these problems:

1. Classify the problem into one the appropriate category; Regular, Context-Free, Turing Decidable, or Undecidable (T1-T4). **(2 marks)**
2. For a problem that is <u>Regular</u>, prove that this is the case by constructing either a DFA, a NFA, or a Regular Expression that accepts the language of the problem. **(5 marks)**
3. For a problem that is <u>Context-Free</u>, do both of the following:
   a. Prove that the problem is not Regular using the pumping lemma, and **(6 marks)**
   b. Prove that the problem is Context-Free by constructing either a PDA or CFG that accepts the language of the problem. **(6 marks)**
4. For a problem that is <u>Turing Decidable</u> do all of the following:
   a. Prove that it is not Context-Free using the pumping lemma for Context-Free grammars, and **(8 marks)**
   b. Prove that the problem is Turing Recognizable by constructing a Turing Machine, and **(10 marks)**
   c. Prove that the Turing Machine constructed is a decider. **(2 marks)**

   <u>Alternatively</u>, instead of a-c above you may choose to prove that a problem is <u>Turing Decidable</u> by reduction to and from a known Turing Decidable problem. **(20 marks)**
5. For a problem that is <u>Undecidable</u>, prove this using reduction from A_{TM}. **(20 marks)**

Problem 1:

The language of binary strings that consist of $1^n0^m1^m0^n$ where m,n>0.

Problem 2:

The wide-field radio-telescope array has been picking up a range of interesting signals which need to be stored, but due to the amount of data coming in only those that are truly relevant should be placed into long-term storage for analysis. A PhD student has developed an algorithm to mark binary strings that may be of interest, but a number of them have found to lack a key part of data, making them useless. It is your job to write a program to scan binary strings suggested by the student's algorithm for this data. Strings which do not contain the key – the sub-string 0010 – should be noted by your program (an 'accept') so that they can be marked as not worth of transmission and storage.

Problem 3:

A conspiracy theorist has decided that Garfield cartoons form a secret code. The idea is that each panel in each cartoon is assigned a 'character number', which is the number of main characters (Garfield, Odie and Jon) who are clearly visible in that panel. The character numbers then form a string, which is then translated into binary and hence into English.

A linguistics expert has pointed out that if such a code were to exist, certain groupings would naturally occur more than any other. In particular, the sequence 011 would occur roughly three times as often as 003 and twice as often as 100.

The problem is to write a program that checks a string of 'character numbers' to see if the expected frequency is upheld.

Problem 4:

A student studying machine perception has written a program that is supposed to classify image instances into ones that fit a specific criteria and those that don't. Unfortunately the program has rejected all images that have been input. His lecturer has found this to be interesting since there is no easily findable bug, and has offered the student at least some marks if he can discover what images the machine does accept. Unfortunately, the student has yet to find such an image and is getting frustrated.

He asks you to write a program to test whether his machine accept any images at all. His lecturer asks that you make your program generic, so future students can also use it to test their programs.

Problem 5:

For a ternary code ($\Sigma = \{a,b,c\}$), find all strings of the form $\{ a^i b^j c^k \}$, $0<k<j<i$.

Problem 6:

A fan of ancient Rome wishes to implement a heuristic test for the Caesar cipher – a simple encryption method that increments every letter in a message by three. The test is to check whether the letter 'h' appears at least three times more often than the letter 'e' in a piece of English text; if so, the text is probably encrypted with a Caesar cipher.

END OF SECTION

Section 2 - Complexity.

Choose one of the problems in this section. For this problem, classify it as being a member of P, NP, NP-Complete or NP-Hard and then prove this classification. When proving membership of NP-Complete or NP-Hard by reduction, use SAT, 3-SAT or 3-COLOR as the problem to reduce to and/or from.

Attempt only one of the questions in this section. If you attempt more than one, only the first question attempted will be marked and any others will be ignored. **(20 marks)**

Problem 1:

An agent from a security agency that must remained un-named requires that your group assist in checking the work of a second group, who are referred only as the S-team. The S-team has created a program that takes a graph that represents a fibre-optic network, and computes a minimal set of nodes (called the 'nodes of interest') in the network such that every link in the network is adjacent to at least one of the nodes. The idea is that the security agency can then install 'appropriate measures' in these nodes in order to have access to all traffic in the area.

Your team's job is to write some software that takes the network (in the form of a graph) and the size of the answer from the S-team's program and checks whether the network contains an appropriate set of 'nodes of interest' that is one smaller than the provided answer. For example, if the S-team's answer has 300 nodes, your program would check if a 299 node answer exists.

Problem 2:

A student is working on a final-year project involving an online gaming scenario called 'Wumpus World'. In this scenario, an intelligent agent must explore a world (which can be modelled as a graph) to locate hidden gold. The first part of the project – identifying dangers – has been completed, and all parts of the world that are to be avoided have been removed from the world (*i.e.*, the graph). The next step is to write a function that plots a course for the agent through the world. Because reducing the amount of moving that the agent needs to do is a requirement, and ideal path to search for the gold would pass through every room exactly once.

This turns out to be more difficult than expected, so the student asks you to help him by writing code that merely checks whether such a path exists for any given graph; if it does not, he plans to use heuristic-based exploration rather than path-finding.

END OF SECTION

Section 3 – Short Answer.

For each of the questions in this section, do the following:

- If asked whether a statement is true or false, state this clearly. You should specifically use one of these two words. If you believe that the statement is both (perhaps with conditions) write "both true and false" or if you believe that it is neither write "neither true nor false".
- If asked to give any other opinion or classification, state it clearly.
- Justify your answer or opinion. Answers without justification will be awarded zero (0) marks.

Question 1:

Two students studying the theory of computing are arguing about the classification of the problem "recognize all strings that contain three or more zeroes, followed by the same number of ones". Classify this problem into Regular, Context-Free or Turing Decidable (or Tier 1-3) and briefly explain your classification.

You do not need to provide a machine or a proof. **(3 marks)**

Question 2:

You have created a DFA that solves a particular problem but you actually need a Regular Expression. You decide to use the formal conversion process, but this process requires a NFA rather than a DFA to start with. Briefly explain how to convert the DFA into an equivalent NFA. **(2 marks)**

Question 3:

The same two students from Question 1 above are having another argument. This time, one claims that a problem is definitely in P while the other states that the problem is in NP. You check their arguments, and it seems that they are both right. Is this possible?

(3 marks)

Question 4:

An aspiring researcher has proved that a problem reduces to and from A_{TM} and also gives an algorithm whose time complexity is polynomial. Assuming her work is correct, does this mean that P = NP? **(3 marks)**

END OF EXAMINATION