Theoretical Foundations of Computer Science

2017 Assignment

This assignment will not be submitted, but the questions in the assignment test will be based directly on the questions below. Note that while some questions may be identical, the general idea is that the concepts covered in the questions are the same. In other words, you need to understand the answers to these questions, not just memorize them.

Section 1 - Classification

For each of the following problems, classify it into a category (Regular, Context-Free, Decidable, Undecidable) and prove its membership of that category. You can use Tier 1-3 and 4+ instead of these categories if you wish.

To prove membership, do the following:

- Regular – provide a DFA, NFA or RE that solves the problem
- Context-Free – provide a proof (preferably using the pumping lemma) that it is not Regular and provide a PDA or CFG that solves the problem
- Turing Decidable – provide a proof (preferably using the pumping lemma for CFGs) that it is not Context-Free and provide a Turing Machine M and prove that M is a decider. If the format of the problem precludes the application of PDAs or CFGs the pumping lemma proof can be omitted and the format problems should instead be used as an argument.
  OR you can prove that a problem is Turing Decidable by reduction to and from a known Turing Decidable problem.
  OR some sensible combination of the two options above.
- Turing Undecidable – prove (using reduction from A_{TM}) that the problem is undecidable

If you cannot prove or disprove membership formally, give a convincing argument. If you need to make assumptions, state these clearly. If you are not sure of a term, make sure that you look up a definition from a reliable source.

PROBLEM 1:

For the language L = {🐼, 🏅, ⚖, ⚽} you are to detect all strings that have every panda getting a medal for football (in other words, the symbol 🐼 is always followed immediately by the symbol 🏅 and the symbol ⚽, in that order) and no reports involve pandas (in other words, the symbol ⚖, is never followed immediately by the symbol 🐼).

PROBLEM 2:

A scientist who has spent his life monitoring termites, decides to turn his attention to ant colony establishment. When a new queen hatches, a group of ants from the existing hive will migrate with her to form a new hive. After some initial observation, the scientist hypothesises that every such migration includes some number of soldier ants (S) and at least twice that number of worker ants (W), in addition to the queen herself (Q). He has never observed eggs

(E) being carried with a migration and hypothesises that this never happens.

You are to determine whether the hypothesis is correct.

PROBLEM 3:

An aerial drone is being used to conduct a study on congestion on the Kwinana Freeway. As it flies, it locates a line of ten or more cars that are in one lane and closely following one another. When such a group is found, it measures the speed and acceleration of each of the cars (as a pair of integers in $^{m}/s$ and $^{m}/s^{2}$)) and records this as a dataset. When the group of cars is reduced to less than 10 cars or the first of these cars leaves the Kwinana Freeway, the drone locates another group. The aim is to take one dataset per minute, although this isn't always possible.

Your job is to write software that will take a dataset as an input and accept if it meets a particular requirement. This requirement is met when a dataset contains one or more cars in the set (call the group Z) have a speed of zero, all of the cars in Z are adjacent to each other, all of the cars before Z have positive acceleration and all cars after Z have negative acceleration.

PROBLEM 4:

After much work, the Wiyrkomi Corporation has begun to produce software systems that are supposed to be able to learn from a user's use of devices such as their smartphone and computer, especially their viewing patterns, interests and online behaviour. The idea is that these learning systems will be improved until they can eventually become working copies of the person that they are learning from. Wiyrkomi is referring to these software systems as GPPs. They are currently implemented on a generic cloud platform (referred to as Heaven), but there some talk of creating customized hardware that incorporates non-Von-Neumann architecture to improve the process.

While this goal is still far away, the Wiyrkomi Corporation is employing your team to write some testing software. Recently, several of the GPPs have been showing worrying agreement on outputs, leading rise to a belief that the learning system has a basic fault that may cause multiple GPPs to converge to an equivalent 'personality'. The programmers in charge strongly deny this, which is why an outside group is needed.

The software should take two GPPs and determine whether, for any possible input, the two will produce identical outputs. For the sake of this question, you may treat each GPP as a black box that accepts finite binary strings and outputs a "yes" or "no" after a finite amount of time.

PROBLEM 5:

A mathematician asks you to design a program that accepts two graphs as input and tests whether these are isomorphic.

PROBLEM 6:

After getting some strange results from a computer program, a research student realises that the program may be having problem with a method that is supposed to categorize an input by either membership or non-membership of a set. Every input she tries is rejected, including ones that aren't supposed to be. Curious, she decides to determine whether every possible input is rejected but also has problem with this.

Your job is to create a general way of checking a method that returns *true* or *false* based on

classifying an input (for the sake of generality consider the input to be simply a binary string and ignore what that string may represent). You test should return *true* if the method always returns *false*, irrespective of input. If there is even a single input for which the method would return *true*, your test should return *false*.

PROBLEM 7:

The controller for the cooling system for a cold storage area changes the temperature level by increasing or decreasing the level of cooling. A thermostat measures the temperature in the room, and sends one of the following signals to the controller:

i. 0 – the temperature is correct
ii. 1 – the temperature is too hot
iii. 2 – the temperature is too cold
iv. 3 – an error has occurred

Based on these signals, the state of the controller becomes one of the following:

i. INC – increase the level of cooling if the room is too hot
ii. DEC – decrease the level of cooling if the room is too cold
iii. MAIN – maintain the current level of cooling
iv. ERR – report an error and stop monitoring the thermostat output

The controller is considered working correctly (acceptable) when not in an error state. You may assume that the cooling can be increased or decreased as much as is required.

PROBLEM 8:

The language of binary strings that contain the sub-string 01^n0 and the sub-string 10^n1 where $0<n$.

PROBLEM 9:

The language of binary strings that contain 1^n0^m but not 1^m0^n where $m > n > 2$.

PROBLEM 10:

For a ternary code ($\Sigma = \{0,1,2\}$), find all strings of the form $0^i1^{i+1}2^{2i}$, $i \geq 0$.

PROBLEM 11:

ES_{TM}, which is the problem of testing a TM *M* to see whether M accepts the empty string.

Section 2 - Complexity

For each of the following problems, classify the problem into one or more time-complexity categories (P, NP, NP-complete, NP-hard). If you cannot prove membership formally, give a convincing argument. If you need to make assumptions, state these clearly. Note that if there is no known classification for an underlying problem I don't expect you to create one, but I would expect you to be able to determine that this is the case and provide references. You may have to look up terms in order to understand some of the questions fully, but I expect you to be able to do this.

When using reduction to show a problem is NP-Complete or NP-Hard, relate the problem to SAT, 3-SAT, or 3-COLOR.

(questions to come during the break)