<u>COT 4210 Quiz #3 Part B: Undecidability, Reducibility 3/25/2021</u>

Regular Start Time: 2:10 pm (EST)
Regular End Time: 2:45 pm (EST)
Regular Late Time: 2:55 pm (EST)

4) (10 pts) Prove that the following language is undecidable:

L_4 = { (<M>, k) | <M> is the encoding of a Turing Machine which accepts all strings of length k}

<u>Solution</u>
We will use proof by contradiction to prove the claim. Assume to the contrary that L_4 is decidable and let a Turing Machine X decide membership in L_4. We describe how to create a Turing Machine Y which decides membership in A_{TM} by using X:

Y(Turing Machine M, String w) {

1. Create a Turing Machine M' which operates as follows: if its input is length |w|, it erases its input, replaces it with w, and then follows all of the steps of M. If its input is not length |w|, M' rejects.
2. We call TM X with input (M', |w|) and accept if X accepts (M', |w|) and reject if X rejects (M', |w|)

}

Since we've created a decider for A_{TM}, which is known to be undecidable, we have arrived at a contradiction. It follows that our initial assumption, that L_4 was decidable is incorrect, which means that L_4 is undecidable, as desired.

Grading Criteria: +2 for assuming decider for L_4
+2 for attempting to write a decider for a known decidable language
+5 for the actual proof
+1 for the conclusion

5) (12 pts) Consider the two following problems:

SAFE-CLASSES = { (S, E) | S is a list of students, and E is a list of pair of students who are not allowed to be in the same class, and there exists a way to split all students in S into two non-empty classes such that the requirements given by E are satisfied }

TWO-COLORABLE = { G | G is an undirected graph such that each vertex in G can be assigned one of two colors, Red or Blue, such that no edge in G connects two vertices that have the same color. }

SAFE-CLASSES is mapping reducible to TWO-COLORABLE. To do the reduction, given an input of students and pairs of students who can't be in the same class, you can create a graph G that is two-colorable if and only if there's a way to split the students in S into two classes satisfying the restriction.

(a) Give an unambiguous algorithmic description of how to take (S, E) and convert it to a corresponding G that completes the mapping reduction successfully.

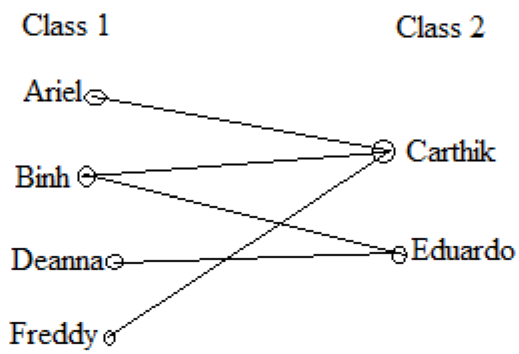
(b) Use your mapping reduction to create a graph for the following input for the problem SAFE-CLASSES:

S = {Ariel, Binh, Carthik, Deanna, Eduardo, Freddy}
E = { (Ariel, Carthik), (Binh, Carthik), (Binh, Eduardo), (Carthik, Freddy), (Deanna, Eduardo) }

(c) Using your graph or otherwise, give a valid listing of two classes of students which proves that the input above belongs to the language SAFE-CLASSES

<u>Solution</u>
(a) For each student in S, create a vertex in the graph G. For each pair of students in E, add an edge in G between the pair of corresponding vertices.

(b) Here is the corresponding drawing for G created by the mapping reduction described in (a):



(c) From the drawing above, we can see that one valid way to split the classes is to have {Ariel, Binh, Deanna and Freddy} in one class and {Carthik, Eduardo} in the other.

Grading Criteria: (a) +2 for ANY valid graph construction
+2 for the correct number of vertices, 1 per student
+2 for adding edges for each pair that can't be together.
(b) +4 for any drawing consistent with the description, give partial as needed
(c) +2 for a valid class split, but I think this might be the only one that works.

7) (3 pts) Who invented the idea of the Turing Machine?

Alan Turing, give to all