CSCI2243-Assignment 1

Assigned August 28, due Wednesday, September 6 at 11PM.

Most of the problems are based on the exercises in Chapter 1 of the textbook. There are 100 points of 'normal' problems. The score for the extra credit problems will be tabulated separately. You may submit a solution to the extra-credit problems any time in September.

1 Syntax of propositional logic

1. Exercise 1 parts (a)-(f). Note that you have to answer the question for both the 'official' grammar and the relaxed syntax rules. *(18 points)*

2. The formula in part (d) of Exercise 1 is a legal formula. Show its derivation using the grammar in the textbook.*(6 points)*

2 Translating natural language into propositional logic

3. Exercise 10. *(10 points)* Use the following variable names (standardizing these will make things easier on the graders):

   - a: The ACT score is at least 29
   - s: The SAT score is at least 1280.
   - r: The high school ranks its graduates.
   - t: The student is in the top 10 per cent of the high school class.
   - g: The core GPA is greater than 3.70.

You should assume that a has the value **false** if the student did not take the ACT, similarly s is false if they did not take the the SAT, t is false if the school does not rank its graduates.

4. Exercise 13.*(8 points)* Use the following variable names:

   - `shoes`: The customer is wearing shoes.
   - `shirt`: The customer is wearing a shirt.
   - `service`: The customer gets served.

   Observe that the structure of this problem is a little bit different from the preceding one. There you had to write a formula using the five variables equivalent to the student receiving automatic admission. Here you do not write a formula equivalent to the customer being served, because there may be other requirements for service. Instead you write a formula involving all three variables that expresses what the sign outside the restaurant intended.

3 Normal forms, equivalence, and identities

5. Exercises 17 and 25 for the formulas in parts (d-g) of Exercise 17. *(16 points for Exercise 17, 16 points for Exercise 25)*

6. Exercise 18. *(12 points)*

4 Binary decision trees.

7. Exercise 32. *14 points* A practical consequence of this problem is that it shows that nested `if-else` structures express *exactly the same thing* as expressions of type `bool`. You need to describe both processes: Going from the decision tree to the propositional formula, and conversely. (HINT: Think DNF for both directions.)

5 Extra credit problems.

8. Exercise 7. *15 extra-credit points* To rephrase the hint given in the problem, our grammar for propositional formulas has only one 'non-

terminal' symbol, `formula`, from which it is possible to derive all formulas. The trick is to add a new symbol, let's call it `special formula`, from which you can derive the formulas that can be combined with others to form larger formulas. As a test to see if you have solved this problem correctly, your new grammar should be able to produce both the formulas in (d) and (e) of Exercise 1, as well as the same formulas with the outer pair of parentheses removed.

9. *25 extra-credit points* Write a Python program that converts a CNF formula into an equivalent DNF formula. You can represent the CNF formula as a list of lists. For example,

$$[[-1, 2, 3], [1, -2], [2, -3]]$$

represent a CNF formula with the three variables p_1, p_2, p_3:

$$(\neg p_1 \vee p_2 \vee p_3) \wedge (p_1 \vee \neg p_2) \wedge (p_2 \vee \neg p_3).$$

The output of the conversion function is likewise a list of lists, but this time, each item in the main list represents the AND of the literals it contains, and the large list represents the OR of these items.

There are two ways to proceed here. You can try to do it *semantically,* by figuring out all the satisfying assignments, or *syntactically*, using just the algebra of propositional formulas.