Logic Project 1: Truth Tables

Adrian Crăciun

`acraciun@info.uvt.ro`

December 11, 2013

The Aim of the Project

Given an arbitrary propositional formula, construct its truth table. Report whether the given formula is a tautology, is satisfiable, or is unsatisfiable. From the truth table, build the disjunctive normal form (DNF) of the formula.

Requirements

- Using the implementation, the user should be able to introduce a propositional formula :

  - files are preferred for the input, to allow the user prior preparation of the input,
  - other input methods (visual forms, etc.) are not restricted, but "live" console input is not encouraged (due to limitations);

- the program should check that the input is a proposition,

- then construct the truth table as described in the lecture;

- for the computation of values under interpretations/truth valuation, a function should be implemented, that can be called explicitly with arguments the formula, and the interpretation, with an option to show all the steps of the computation,

- the program should then print the truth table (on the screen and/or in a file - latter preferred),

- and give an analysis of the formula: whether it is a tautology, satisfiable or unsatisfiable,

- the program should also provide a disjunctive normal form equivalent of the input formula.

Programming Language

C or Python strongly suggested. However, other languages will be allowed[1].

---

[1] This should be discussed with me before the start of the project and properly motivated

Delivery of the Project

The project will be delivered in the following form:

1. A paper, describing the implementation, that should contain the following:

   - the statement of the problem,
   - user manual, i.e. how to use the program,
   - a description of the solution (structure of the program, functions, data structures),
   - an argument (proof) that the program is correct,
   - a test suite[2] for the program, i.e. examples of inputs, including nontrivial ones,
   - test runs (e.g. screen captures, or printouts), with timing,
   - conclusions (whatever the author(s) experience was).

2. Annexes to this paper, containing:

   - the license of the program (see `http://en.wikipedia.org/wiki/Software_licenses`); also provide a reason for the choice of license,
   - the code of the program,
   - the installation manual for the program.

   NOTE. The paper and annexes will be delivered in .pdf format, by email, or made available online.

3. The program (source) should be sent by email, or made available online.

4. The program will be demonstrated at the University (arrangements to be discussed).

Working in Teams

Working in teams (max. **3** people) is allowed, and even encouraged. However, this should be a learning experience. Therefore, people with experience in programming are encouraged to team up with the people which are less familiar with this activity. At delivery, any of the team members should be able to answer any question connected to the project.

---

[2]The test suite should include all homeworks (where it applies)