CME 2101 Project-Based Learning III
Project-2 - Logic Expression Simplification Tool

In this project you are expected to design a simplification tool for logic expressions. You will write a windows application to design this tool by using Java. The main purpose is to simplify given expressions which are either in Boolean format (Sum of Products) or as truth table. The functions will be 1 output and up to 4 inputs. The tool should have the following specifications:

- Reading the files that include expressions. File types will be given as .tt (for truth table - see example 1) or .be(Boolean expression - see example 2)
- Accepting logic expression inputs from GUI as a truth table or an equation.
- Exporting simplified expression as files (.tt or .be).
- For the simplification two methods should be used. One of them is Boolean Algebra in which, used steps should be shown. The other is Karnaugh Map in which groups that used for simplification should be shown.
- Showing any function as a truth table, equation or Karnaugh map.
- Displaying an alternative simplified functions for each if they exist.
- Showing corresponding expression term on the Karnaugh Map as highlighted and vice versa.
- JavaFX library will be used for the project.
+ **Bonus (5 points)**: Showing simplified expression as a multi-level gate diagram.

Truth Table Format in *.tt* files:
```
A,B,C;F
0,0,0;0
0,0,1;1
…
1,1,1;0
```

Example 2 - Logic Expression Format: F = A'.B'.C + A'.B.C' + A.B'.C + A.B'.C'

Rules for Boolean Simplification

Idempotent Law	A.A = A	A+A = A
Involution Law	(A')' = A	
Associative Law	(A.B).C = A.(B.C)	(A+B)+C = A+(B+C)
Commutative Law	A.B = B.A	A+B = B+A
Distributive Law	A.(B + C) = A.B + A.C	A+(B.C)=(A+B).(A+C)
Identity Law	A.0 = 0 A.1 = A	A + 1 = 1 A + 0 = A
Complement Law	A.A' = 0	A + A' = 1
Absorptive Law	A(A+B) = A	A+(A.B) = A
Derived Laws	A+A'.B = A+B	(A+B).(A+C) = A+B.C
DeMorgan's Law	(A.B)' = A'+B'	(A+B)' = A'.B'

Good Luck.

Your project outcome should include a report as well as a working computer program. In your report, you should explain indexing method, data structures, algorithms and relevance measure used by your programs. The programs will be checked for the plagiarism; similar codes will get zero points.

Table 1. Weekly Schedule

Week	Date	Type	Description
1	22.11.2016 Tuesday	Lecture	Introduction to Project and GUI Design
	24.11.2016 Thursday	**Lab 1**	**GUI Design in JavaFX**
2	29.11.2016 Tuesday	Discussion	GUI and Class Design
	01.12.2016 Thursday	**Milestone 1**	**GUI, Class Design, File Operations**
3	06.12.2016 Tuesday	Discussion	Boolean Algebra Structure
	08.12.2016 Thursday	Discussion	Boolean Algebra Simplification
4	**13.12.2016 Tuesday**	**Milestone 2**	**Boolean Algebra**
	15.12.2016 Thursday	Discussion	Karnaugh Map Structure
5	20.12.2016 Tuesday	Discussion	Karnaugh Map Simplification
	22.12.2016 Thursday	**Milestone 3**	**Karnaugh Map**
6	27.12.2016	Discussion	Testing, reporting, fine tuning, etc.
	29.12.2016	Presentation	Video

Table 2. Grading Policy

Item	Expectations . Requirements	Perception	
Milestone 1	Graphical User Interface (50%) Class Design (20%) File Operation (30%)	5%	
Milestone 2	Data structure for Boolean Expression (30%) Implementing Boolean Algebra Rules (30%) Boolean Algebra Simplification (40%)	10%	
Milestone 3	Data structure for Karnaugh Map (30%) Implementing Karnaugh Map (30%) Karnaugh Map Simplification (40%)	10%	Group (%75) * Contribution Factor
Functionality	● Graphical User Interface (10%) ● Class Design (10%) ● Reading and exporting expressions (10%) ● Generating Truth Table (10%) ● Boolean Algebra Rules (10%) ● Simplification using Boolean Algebra (20%) ● Karnaugh Map Displaying (10%) ● Simplification using Karnaugh Map (20%)	40%	
Project report		10%	
Attendance		5%	
Active participation and contribution to sessions		10%	
Video Presentation		10%	

Attendance notice: If you don't attend more than 3 meetings, your overall project score will be graded with zero. Non-attendance of 1, 2 or 3 meetings will be punished by reduced marks in your final score for the course.