CS 270 - Lab 15

M. Boady, B. Char, J. Johnson, G. Long, S. Earth

1 Introduction

You may work in teams of **one** or **two** students. Submit one copy for the entire group.

Write your answers on this lab sheet. Only what is written on this lab sheet will be graded.

This lab is due at the end of the class period. You may not continue to work on it once class has ended.

This lab contains 4 questions.

Grading

- 25 points - Putting everyones names on this page
- 20 points - Earned for each correct question (Answer is fully correct)
- 5 points - Earned for **partial credit** any question

No additional point amounts can be earned. You cannot earn 7 points on a question for example.

The maximum score for a lab is 100. If you get everything correct, that adds up to 105 points but will be reduced to 100.

A question will be marked correct as long as it covers all requirements of the question. It does not need to be perfect, but must be fully correct. A single typo or very minor issue where the intention is clear and all requirements are met would still earn full points.

We want you to complete questions fully, not try to earn partial credit on multiple questions. You may ask your Professor/Course assistant questions during lab.

Labs must be done in the presence of an instructor and/or course assistant or credit will not be given.

Partners should alternate each class day which person is physically typing and submitting the lab.

Do not split up the problems or you risk not finishing on time due to the cumulative nature of the questions.

Enter the name of the student in the group

Member 1 (submitter): Elan Rubin

Member 2: Carmelo Wheeler Timmons

Question 1 :

A Boolean Expression is in **Conjunctive Normal Form** if it is a **conjunction** of one or more clauses, where a clause is a **disjunction** of **literals** and it is in **Negation Normal Form**.

An expression in **CNF** form must meet the following requirements.

- Negation is only applied to variables
- Only binary operators allowed are $\land$ and $\lor$
- All conjunctions are above all disjunctions

For each of the below expressions determine if it is in CNF form.

If it is in CNF form write **CNF**.

If it is not in CNF form write $\neg$ **CNF** otherwise.

An expression is in CNF form **if and only if** it does not break any of the above rules. For example, $A \lor B$ is in CNF. It does not break any of the three rules.

Grading: You will still receive full credit for this question if you get 1 wrong.

(a) X

(a) CNF

(b) $\neg A$

(b) CNF

(c) $A \lor (X \land Y \land A)$

(c) ¬CNF

(d) $(A \lor B) \land (X \lor \neg\neg A)$

(d) ¬CNF

(e) $A \land \neg(B \lor X)$

(e) ¬CNF

(f) $X \lor Y \lor Z$

(f) CNF

(g) $(A \land B) \lor A$

(g) ¬CNF

(h) $(A \lor B) \land (X \lor \neg B)$

(h) CNF

(i) $\neg(A \implies B)$

(i) ¬CNF

(j) $(A \lor B \lor \neg C) \land (\neg A \lor \neg B \lor C) \land X$

(j) CNF

Question 2 :

You can change any non-CNF expression into a CNF expression using the following rules.

This means any expression can be written as CNF.

$A \iff B = (A \implies B) \wedge (B \implies A)$	Def of BiCond
$A \implies B = \neg A \vee B$	Def of Implies
$\neg(A \vee B) = \neg A \wedge \neg B$	DeMorgan Or
$\neg(A \wedge B) = \neg A \vee \neg B$	DeMorgan And
$\neg\neg A = A$	Double Neg Elim.
$A \vee (X \wedge Y) = (A \vee X) \wedge (A \vee Y)$	Distribution

An example simplification is given below.

1.	$\neg(A \implies (\neg B))$	Premise
2.	$\neg((\neg A) \vee (\neg B))$	By Def of Implies
3.	$(\neg(\neg A)) \wedge (\neg(\neg B))$	By DeMorgan Or
4.	$A \wedge B$	By Double Neg Elim.

Convert $\neg(A \iff B)$ to CNF form. Label each line with the rule you used.

Hint: Be careful of parenthesis! It is safer to write them all out. For example $(\neg A) \vee B$ instead of $\neg A \vee B$ to avoid mistakes.

¬(A⇐⇒B) premise
¬((A⇒B)∧(B⇒A)) def of bicond
¬((((¬A)∨B) ∧ ((¬B)∨A))) def of implies (applied to each)
(¬(¬A) ∧ ¬B) ∨ (¬(¬B) ∧ ¬A) DeMorgan Or
(A ∧ ¬B) ∨ (B ∧ ¬A) double neg elim (applied to each)
(A ∨ B) ∧ (A ∨ ¬A) ∧ (¬B ∨ B) ∧ (¬B ∨ ¬A) distribution

Question 3 :

SAT in Browser

Complete this Section using

`https://www.msoos.org/2013/09/minisat-in-your-browser/`

MiniSat has a simple language based on CNF. It is called DiMACS.

It only accepts CNF expressions.

Example:

```
p cnf 4 2
1 2 0
-3 4 0
```

Translates to $(x_1 \vee x_2) \wedge (\neg x_3 \vee x_4)$

The first line says that the expression has 4 variables and 2 clauses.

Each Clause ends with a 0. You cannot use 0 as a variable.

When this problem is solved by MiniSat it prints out.

```
SATISFIABLE
v -1 2 -3 -4 0
```

This tells us to set x_1=F, x_2=T, x_3=F, x_4=F to make the statement true.

Use MiniSat to find a Satisfying assignment for the below expression.

$$(x_1 \vee x_2 \vee x_3) \wedge \neg x_2 \wedge \neg x_3$$

(a) Give the DiMACs code for this expression.

```
p cnf 3 3
1 2 3 0
-2 0
-3 0
```

(b) Run the code in DiMacs. Write the solution provided by MiniSAT below. (Something like `v -1 2 -3 4 ...`)

v 1 -2 -3 0

(c) What is the setting of variables to make the expression true? (Something like x_1=T, ...)

x1 = T, x2 = F, x3 = F

(d) Verify the SAT Solver wast correct. Evaluate $(x_1 \vee x_2 \vee x_3) \wedge \neg x_2 \wedge \neg x_3$ using the results of the SAT solver.

(T ∨ F ∨ F) ∧ ¬F ∧ ¬F
(T) ∧ T ∧ T
T

correct

Question 4 :

In this question, you will use the SAT solver to solve a problem.

Suppose a and b are integers and A is a proposition that a is even, and B is a proposition that b is even.

Using only two variables, express in DiMACs that $a + b$ is even. (*Hint*: it may take multiple clauses)

(a) In DiMACs we use numbers to represent variables. List what each of your numbers being true means. (For example, if 2 is true that means the ocean is blue.)

A being true means it's even
B being true means it's even

A being false means it's odd
B being false means it's odd

(b) Write your DiMacs expression for $a + b$ is even. (Remember it must be in CNF form $(\cdots \vee \cdots) \wedge (\cdots \vee \cdots) \wedge \cdots$)

CNF: (A ∧ B) ∨ (¬A ∧ ¬B)
CNF simplified: (A ∨ ¬B) ∧ (¬A ∨ B)
DiMACs:
p cnf 2 2
1 -2 0
-1 2 0

(c) What is the solution provided by the SAT solver?

s SATISFIABLE
c conflicts: 0
v -1 -2 0

(d) Explain in plain English the meaning of the solution. (Example: -4 that means bears are black.)

If you're answer doesn't seem right, go back to partB and make certain your DIMACs clauses are all correct

-1 means A is not even
-2 means B is not even

if two numbers that are both not even are added, their sum is a number which is even