# CSE112 Artificial Intelligence， Week 10 2019

**Exercises And Tutorial Questions**

≡

## Exercise 1

Transform the following formula into an equivalent formula in CNF: (¬p  ¬q)  (p  q)

≡(p¬q) (pq)

≡(p¬q) (pq)

≡(pq) (pq)

≡ (p pq) q  pq 

≡ (p q) q  p

≡ p q

## Exercise 2

Compute the CNF of ((P  Q) P)  P

((P  Q) P)  P

≡ ((¬P  Q) P)  P

≡ (¬ (¬P  Q) P)  P

≡ ((P  ¬Q) P)  P

≡ ¬ ((P  ¬Q) P)  P

≡ (¬ (P  ¬Q) ¬P)  P

≡ ((¬ P  Q) ¬P)  P

≡ ((¬ P  Q) ¬P)  P

≡ ((¬ P ¬P)  (Q¬P))  P

≡ (¬ P  (Q¬P))  P

≡ Q¬P

## Exercise 3

This question is about testing whether a CNF (Conjunctive Normal Form) formula is a tautology. You will show that it is quite easy to test if a CNF formula is a tautology, in contrast with testing whether a CNF formula is satisfiable, which is not so easy.

1. Answer whether each of the following CNF formulas is a tautology (no justification required):

1.(a∨ ¬b∨c)

2.(a∨b∨c∨ ¬b)

3.(b∨ ¬b)∧(¬c∨d)

1. Argue that α∧β is a tautology if and only if α is a tautology and β is a tautology.

true

1. Argue that a CNF clause is a tautology if and only if it contains x and ¬x for some variable x.

true

1. Give an algorithm to check if a CNF formula is a tautology. (Give a brief high- level description, just in words.)

**Exercise 4** Assume that the following sentences are in our Knowledge Base (A denotes the negation of A):

A

A => B  C

B => D  E

C => D  E  F

Using the inference rules that we studied in class for propositional logic, prove that “F is true”. When you derive F, specify exactly the sequence of inference rules that you used.

A 1

A => B  C 2

B => D  E 3

C => D  E  F 4

B  C 5 (1,2)

B 6 (5)

C 7 (5)

D  E 8 (3,6)

D 9 (8)

E 10 (8)

D  E  F 11 (4,9,10)

F 12 (4,11)

## Exercise 5

An argument of propositional logic is of the form 1 ,… ,n:  , where i, i are all expressions of propositional logic. The expressions before the colon are the premises of the argument, and the one after the colon is its conclusion. An argument is valid if and only if there is no possible assignment of truth values to atomic propositional symbols such that the premises are all true and the conclusion false.

Using a truth table, determine whether the following arguments are valid or invalid:

* 1. (A  B) A, B A : A  B

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| A | B | A  B | (A  B) A | B A | A  B |
| 0 | 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 |

invalid

* 1. A  (B  C), B  C, C  A : A

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| A | B | C | B  C | A  (B  C) | B  C | C  A |
| 0 | 0 | 0 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 |

invalid

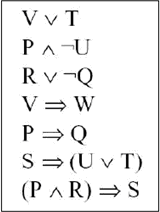
**Exercise 6** Prove by means of natural deduction If John is not married he is a bachelor. (P Q) John is not a bachelor. (Q)

Therefore, he is married. (P)

## Exercise 7

We have a knowledge base (KB) as shown below,

(a) In the space provided to the right of the KB, rewrite the KB in conjunctive normal form.



(b). In this KB, show an example of how -Elimination can be used to add a new sentence to the KB.

1. In this KB, show an example of how Modus Ponens can be used to add a new sentence to the KB.
2. In the CNF KB from above, show an example of how resolution can be used to add a new sentence to the KB.

## Exercise 8

Propositional Logic: Convert each of the following propositional sentences into conjunctive normal norm (CNF) unless it is already in CNF:



# Sample Rules of Inference

### Modus Ponens {a=> f3, a} I- f3

* + And Elimination: {o: A f3} I- a; {a A f3} l-f3
  + And Introduction: {a, B} I- a/\ f3
  + Or introduction: {a} I- av f3
  + Double negation Elimination: {-,--,a} 1- a
  + Implication Elimination: {a=> f3} 1- -,av f3
  + Unit resolution: {av f3, -,f3} I- a
  + Resolution: {av f3, -,f3v y} I- av y