**Exercise 1, Chapter 3 (3.1) + Exercise 2, Chapter 4 (4.2)**

Predicates:

Alpinist(x) : x is a member of the Alpine Club

Mountain(x) : x is a mountain climber

Skier(x) : x is a skier

LikesRain(x) : x likes rain

LikesSnow(x) : x likes snow

Entailments:

∀x[Alpinist(x) ∧ ¬Skier(x) ⊃ Mountain(x)] or [¬Alpinist(x), Skier(x), Mountain(x)]

∀x[Mountain(x) ⊃ ¬LikesRain(x)] or [¬Mountain(x), ¬LikesRain(x)]

∀x[¬LikesSnow(x) ⊃ ¬Skier(x)] or [LikesSnow(x), ¬Skier(x)] in *CNF*

Knowledge Base:

Alpinist(Tony)

Alpinist(John)

Alpinist(Mike)

LikesRain(Tony)

LikesSnow(Tony)

¬LikesRain(Mike)

¬LikesSnow(Mike)

Mountain(Tony) ⊃ ¬Mountain(Mike) or [¬Mountain(Tony), ¬Mountain(Mike)]

Skier(Tony) ⊃ ¬Skier(Mike) or [¬Skier(Tony), ¬Skier(Mike)] in *CNF*

**(3.1)**

Question:

1. Prove that ∃x[Alpinist(x) ∧ Mountain(x) ∧ ¬Skier(x)]

Proof:

1. (∀x[¬LikesSnow(x) ⊃ ¬Skier(x)]) ⊃ (¬LikesSnow(Mike) ⊃ ¬Skier(Mike))

// Mike is not a skier!

(∀x[Alpinist(x) ∧ ¬Skier(x) ⊃ Mountain(x)]) ⊃ (Alpinist(Mike) ∧ ¬Skier(Mike) ⊃ Mountain(Mike)) // Mike is a Mountain Climber!

Alpinist(Mike) // Mike is a member of the Alpine Club

Alpinist(Mike) ∧ Mountain(Mike) ∧ ¬Skier(Mike)

Hence, ∃x[Alpinist(x) ∧ Mountain(x) ∧ ¬Skier(x)]. x is Mike!

**(4.2)**

Question:

Prove that ∃x[Alpinist(x) ∧ Mountain(x) ∧ ¬Skier(x)] using Resolution with Answer Extraction

Proof:

[¬Alpinist(x), Skier(x), Mountain(x)] [¬Alpinist(x), ¬Mountain(x), Skier(x), A(x)]

[LikesSnow(x), ¬Skier(x)] [¬Alpinist(x), Skier(x), A(x)]

[Alpinist(Mike)] [¬Alpinist(x), LikesSnow(x), A(x)]

x/Mike

[¬LikesSnow(Mike)] [LikesSnow(Mike), A(Mike)]

[A(Mike)]

Hence, Mike is the x that proves that ∃x[Alpinist(x) ∧ Mountain(x) ∧ ¬Skier(x)].