

Washington State University  
School of Electrical Engineering and Computer Science  
Fall 2019

CptS 440/540 Artificial Intelligence

**Homework 6**

Due: October 10, 2019 (11:59pm)

**General Instructions:** Put your answers to the following problems into a PDF document and submit as an attachment under Content → Homework 6 for the course CptS 440 Pullman (all sections of CptS 440 and 540 are merged under the CptS 440 Pullman section) on the Blackboard Learn system by the above deadline. Note that you may submit multiple times, but we will only grade the most recent entry submitted before the above deadline.

1. Consider the following logic problem:

All people who like computers also like coding.

All people who like coding and like chess will learn AI.

For all people that learn AI, there is at least one company that will hire them.

All people who are hired by some company will be rich and famous.

a. We will solve this problem using first-order logic. First, show one first-order logic sentence for each of the first four sentences in the above problem. You may only draw from the following first-order predicates.

- Like( $x$ , Chess)
- Like( $x$ , Computers)
- Like( $x$ , Coding)
- Learn( $x$ , AI)
- Hire( $x$ ,  $y$ ) – which means company  $x$  hires person  $y$
- Rich( $x$ )
- Famous( $x$ )

(1)  $\forall x \text{ (Like}(x, \text{Computers}) \Rightarrow \text{Like}(x, \text{Coding}))$

(2)  $\forall x \text{ ((Like}(x, \text{Coding}) \wedge \text{Like}(x, \text{Chess})) \Rightarrow \text{Learn}(x, \text{AI}))$

(3)  $\forall x \text{ (Learn}(x, \text{AI}) \Rightarrow \exists y \text{ (Hire}(y, x)))$

(4)  $\forall x \text{ (\exists y (Hire}(y, x)) \Rightarrow \text{Rich}(x) \wedge \text{Famous}(x))$

b. Convert each of the four sentences from part (a) into Conjunctive Normal Form (CNF). You may just show the final result for each sentence; no need to show the intermediate steps. Number each clause. We will refer to these clauses as the knowledge base (KB) below.

C1:  $\neg \text{Like}(x, \text{Computers}) \vee \text{Like}(x, \text{Coding})$   
 C2:  $(\neg \text{Like}(x, \text{Coding}) \vee \neg \text{Like}(x, \text{Chess})) \vee \text{Learn}(x, \text{AI})$   
 C3:  $\neg \text{Learn}(x, \text{AI}) \vee \text{Hire}(F(x), x)$   
 C4:  $(\neg \text{Hire}(y, x) \vee \text{Rich}(x)) \wedge (\neg \text{Hire}(y, x) \vee \text{Famous}(x))$

- c. To the KB from part (b), add the two facts: “Larry likes computers” and “Larry likes chess”. Using this augmented KB, perform a resolution proof by refutation to prove “Rich(Larry)”. In your proof, be sure to do the following:

- For each resolution step, show the numbers of the two clauses used, the resulting clause, any variable substitutions resulting from unifying the complementary literals, and then number the resulting clause.
- Be sure to use unique variables (e.g.,  $x_1$ ,  $x_2$ ,  $y_1$ , etc.) for each use of a clause from the KB.
- Remember: each resolution step can only be done with two clauses at a time and can eliminate only one literal from each clause.

Cc5:  $\text{Like}(\text{Larry}, \text{Computers})$   
 Cc6:  $\text{Like}(\text{Larry}, \text{Chess})$   
 Cc7:  $\neg \text{Rich}(\text{Larry})$

Cc7 + C4( $x_1$ ):  $\{ x_1 / \text{Larry} \}$   
 Cc8:  $(\neg \text{Hire}(y_1, \text{Larry}))$

Cc8( $y_1$ ) + C3( $x_2$ ,  $F(x_2)$ ):  $\{ x_2 / \text{Larry}, y_1 / F(x_2) \}$   
 Cc9:  $\neg \text{Learn}(\text{Larry}, \text{AI})$

Cc9 + C2( $x_3$ ):  $\{ x_3 / \text{Larry} \}$   
 Cc10:  $\neg \text{Like}(\text{Larry}, \text{Coding}) \vee \neg \text{Like}(\text{Larry}, \text{Chess})$

Cc10 + Cc6:  
 Cc11:  $\neg \text{Like}(\text{Larry}, \text{Coding})$

Cc11 + C1( $x_4$ ):  $\{ x_4 / \text{Larry} \}$   
 Cc12:  $\neg \text{Like}(\text{Larry}, \text{Computers})$

Cc12 + Cc5:  
 Cc13: Empty clause.

Hence,  $\text{Rich}(\text{Larry})$  is true.

- d. To the KB from part (b), add the fact: “There exists someone who learns AI.” Be sure to first convert this fact to CNF. Using this augmented KB, perform a resolution proof by refutation to prove “ $\exists x \text{ Famous}(x)$ ”, following the guidelines in part (c).

KB :-

C1:  $\neg \text{Like}(x, \text{Computers}) \vee \text{Like}(x, \text{Coding})$

C2:  $(\neg \text{Like}(x, \text{Coding}) \vee \neg \text{Like}(x, \text{Chess})) \vee \text{Learn}(x, \text{AI})$

C3:  $\neg \text{Learn}(x, \text{AI}) \vee \text{Hire}(F(x), x)$

C4:  $(\neg \text{Hire}(y, x) \vee \text{Rich}(x)) \wedge (\neg \text{Hire}(y, x) \vee \text{Famous}(x))$

C5:  $\text{Learn}(F1, \text{AI})$

C6:  $\neg \text{Famous}(x)$

$C6(x1) + C4(x2, y1): \{ x1 / x2 \}$

C7:  $\neg \text{Hire}(y1, x2)$

$C7(x2, y1) + C3(x3): \{ x2 / x3, y1 / F(x3) \}$

C8:  $\neg \text{Learn}(x3, \text{AI})$

$C8(x3) + C5: \{ x3 / F1 \}$

C9: Empty clause.

Hence, “ $\exists x \text{ Famous}(x)$ ” is true.

2. *CptS 540 Students Only.* Create an input file for the Vampire theorem prover that can be used to solve Problem 1c. Include your input file and the Vampire output in your Homework 6 submission. You can include them as separate text files or cut-and-paste them into the document containing the rest of your solution. You can download the Vampire theorem prover from <https://vprover.github.io/>. There is a Linux binary available there that runs on the ssh1-ssh10 servers that all grad students have access to. You can run it on your own machine if you prefer or build it from source. For your reference, below is the Vampire input file for the “criminal(west)” proof demonstrated in class.

```
fof(a1, axiom,
    ! [X,Y,Z] : ((american(X) & weapon(Y) & sells(X,Y,Z) & hostile(Z))
    => criminal(X)).
```

```
fof(a2, axiom,
    enemy(nono,america)).
```

```
fof(a3, axiom,
```

```

? [X] : (owns(nono,X) & missile(X))).

fof(a4, axiom,
  ! [X] : ((missile(X) & owns(nono,X)) => sells(west,X,nono))).

fof(a5, axiom, american(west)).

fof(a6, axiom,
  ! [X] : (missile(X) => weapon(X))).

fof(a7, axiom,
  ! [X] : (enemy(X,america) => hostile(X))).

fof(c1, conjecture, criminal(west)).

```

Answer:

Input file:

```

fof(a1, axiom,
  ! [X] : (like(X, computers) => like(X, coding))).

fof(a2, axiom,
  ! [X] : ((like(X, coding) & like(X, chess)) => learn(X, ai))).

fof(a3, axiom,
  ! [X] : (learn(X, ai) => ? [Y] : (hire(Y, X)))).

fof(a4, axiom,
  ! [X] : (
    ? [Y] : (hire(Y, X)) => (rich(X) & famous(X)))).

fof(a5, axiom, like(larry, computers)).

fof(a6, axiom, like(larry, chess)).

fof(c1, conjecture, rich(larry)).

```

Output:

```
% Refutation found. Thanks to Tanya!
% SZS status Theorem for problem
% SZS output start Proof for problem
1. ! [X0] : (like(X0,computers) => like(X0,coding)) [input]
2. ! [X0] : ((like(X0,chess) & like(X0,coding)) => learn(X0,ai))
   [input]
3. ! [X0] : (learn(X0,ai) => ? [X1] : hire(X1,X0)) [input]
4. ! [X0] : (? [X1] : hire(X1,X0) => (famous(X0) & rich(X0)))
   [input]
5. like(larry,computers) [input]
6. like(larry,chess) [input]
7. rich(larry) [input]
8. ~rich(larry) [negated conjecture 7]
9. ~rich(larry) [flattening 8]
10. ! [X0] : (? [X1] : hire(X1,X0) => rich(X0)) [pure predicate
    removal 4]
11. ! [X0] : (like(X0,coding) | ~like(X0,computers)) [ennf
    transformation 1]
12. ! [X0] : (learn(X0,ai) | (~like(X0,chess) | ~like(X0,coding)))
    [ennf transformation 2]
13. ! [X0] : (learn(X0,ai) | ~like(X0,chess) | ~like(X0,coding))
    [flattening 12]
14. ! [X0] : (? [X1] : hire(X1,X0) | ~learn(X0,ai)) [ennf
    transformation 3]
15. ! [X0] : (rich(X0) | ! [X1] : ~hire(X1,X0)) [ennf transformation
    10]
16. ! [X0] : (? [X1] : hire(X1,X0) => hire(sK0(X0),X0)) [choice
    axiom]
17. ! [X0] : (hire(sK0(X0),X0) | ~learn(X0,ai)) [skolemisation
    14,16]
18. ~like(X0,computers) | like(X0,coding) [cnf transformation 11]
19. ~like(X0,chess) | learn(X0,ai) | ~like(X0,coding) [cnf
    transformation 13]
20. hire(sK0(X0),X0) | ~learn(X0,ai) [cnf transformation 17]
21. ~hire(X1,X0) | rich(X0) [cnf transformation 15]
22. like(larry,computers) [cnf transformation 5]
23. like(larry,chess) [cnf transformation 6]
24. ~rich(larry) [cnf transformation 9]
25. like(larry,coding) [resolution 18,22]
26. ~learn(X0,ai) | rich(X0) [resolution 20,21]
27. learn(larry,ai) | ~like(larry,coding) [resolution 19,23]
28. learn(larry,ai) [subsumption resolution 27,25]
29. rich(larry) [resolution 26,28]
```

```
30. $false [subsumption resolution 29,24]
% SZS output end Proof for problem
% -----
% Version: Vampire 4.2.2 (commit e1949dd on 2017-12-14
18:39:21 +0000)
% Termination reason: Refutation

% Memory used [KB]: 4733
% Time elapsed: 0.015 s
% -----

% -----
```