Artificial Intelligence Solution Set 3

Problem 1:

Using propositional logic, prove (D) from (A,B,C):

A. $P \Rightarrow (Q \Leftrightarrow R)$

B. $\neg(Q \Leftrightarrow R)$

C. $(S \wedge Q) \Rightarrow P$

D. $\neg P \wedge (S \Rightarrow \neg Q)$

Answer: Transforming A,B,C and the negation of D to CNF gives the following clauses:

A.1 $\neg P \vee \neg Q \vee R$.

A.2 $\neg P \vee Q \vee \neg R$.

B.1 $Q \vee R$

B.2 $\neg Q \vee \neg R$

C. $\neg S \vee \neg Q \vee P$.

D.1 $P \vee S$.

D.2 $P \vee Q$.

One resolution proof (there are many) then proceeds as follows:

E. $Q \vee \neg R$ (D.2 + A.2, factored)

F. Q. (E + B.1)

G. $\neg R$ (F+B.2).

H. $\neg Q \vee P$. (C+D.1, factored)

I. P (H+D.2, factored)

J. $\neg Q \vee R$ (I+A.1).

K. R (J+F)

L. $\emptyset$. (K+G).

Problem 2: Trace the workings of the Davis-Putnam algorithm in finding a valuation satisfying (A-E) below. Assume that at each choice point, the algorithm picks atoms in alphabetical order, and tries the assignment "true" before the assignment "false".

A. $P \Rightarrow Q$

B. $Q \Rightarrow \neg(R \wedge S)$.

C. $(P \wedge W) \Rightarrow (R \wedge S)$.

D. $\neg$ W $\Rightarrow$ (R $\wedge$ S)

E. R $\Rightarrow$ P.

Answer: Converting these to CNF gives the following clauses.

A. $\neg$P $\vee$ Q.

B. $\neg$Q $\vee$ $\neg$R $\vee$ $\neg$S.

C.1 $\neg$P $\vee$ $\neg$W $\vee$ R

C.2 $\neg$P $\vee$ $\neg$W $\vee$ S

D.1 W $\vee$ R

D.2 W $\vee$ S

E. $\neg$R $\vee$ P.

Let STATE0 be the above set of clauses. Since there are no singleton clauses, we try the assignment P=TRUE. This gives us the new set of clauses, STATE1.

A. Q.

B. $\neg$Q $\vee$ $\neg$R $\vee$ $\neg$S.

C.1 $\neg$W $\vee$ R

C.2 $\neg$W $\vee$ S

D.1 W $\vee$ R

D.2 W $\vee$ S

Since A is a singleton clause, we assign Q := TRUE, giving the new set of clauses STATE2:

B. $\neg$R $\vee$ $\neg$S.

C.1 $\neg$W $\vee$ R

C.2 $\neg$W $\vee$ S

D.1 W $\vee$ R

D.2 W $\vee$ S

Since there are no singleton clauses, we try the assignment R=TRUE, giving the new set STATE3:

B. $\neg$S.

C.2 $\neg$W $\vee$ S

D.2 W $\vee$ S

Since B is a singleton clause we assign S=FALSE, giving the new set STATE4

C.2 $\neg$W

D.2 W

Since C.2 is a singleton clause we assign W=FALSE, giving the new set STATE5

D.2 empty.

Thus, this branch of the search has failed. We return to the last choice point STATE2 and try the assignment R=FALSE, giving state STATE6

C.1 $\neg$W

C.2 $\neg$W $\vee$ S

D.1 W

D.2 W $\vee$ S

Since C.1 is a singleton clause, we assign W=FALSE, given state STATE7

D.1 empty

D.2 S

So this branch has also failed. So we return to state STATE0 and try the assignment P=FALSE, giving STATE8:

B. $\neg$Q $\vee$ $\neg$R $\vee$ $\neg$S.

D.1 W $\vee$ R

D.2 W $\vee$ S

E. $\neg$R

Since E is a singleton clause, we can assign R=FALSE, giving STATE9:

D.1 W

D.2 W $\vee$ S

Since D.1 is a singleton clause, we can assign W=TRUE, given the state with no clauses. Thus, we have found a satisfying assignment: P=FALSE, R=FALSE, W=TRUE. The value of S does not matter.

Problem 3: Consider a universe whose entities are stores, product (e.g. "cabbage", "Can of Coke", etc.) and items (some particular head of cabbage or can of coke.) Let $\mathcal{L}$ be the first-order language with the following non-logical symbols:

- at(I, S) — Predicate: Item I is at store S.
- carry(S, P) — Predicate: Store S carries product P.

- stock(S, P) — Predicate: Product P is in stock at store S.
- inst(I, P) — Predicate: Item I is an instance of product P. B
- superxxx, deli94, cokecan101, canofcoke, tomato — Constants.

State the following sentences in $\mathcal{L}$:

A. Product P is in stock at store S if and only if some instance of P is at store S.

B. If S does not carry product P, then P is not in stock at S.

C. SuperXXX carries every product that Deli94 does.

D. CokeCan101 is at Deli94.

E. CokeCan101 is an instance of CanOfCoke.

F. Tomatoes are out of stock at Deli94.

G. SuperXXX carries CanOfCoke.

H. CokeCan101 is not an instance of a tomato.

Answer:

A. $\forall_{P,S}$ stock$(S, P) \Leftrightarrow \exists_I$ inst$(I, P) \wedge$ at(I, S)

B. $\forall_{S,P}$ $\neg$carry$(S, P) \Rightarrow \neg$stock(S, P).

C. $\forall_P$ carry(deli94,P) $\Rightarrow$ carry(superxxx,P).

D. at(cokecan101,deli94).

E. inst(cokecan101,canofcoke)

F. $\neg$stock(deli94,tomato)

G. carry(superxxx.canofcoke).

H. $\neg$inst(cokecan101,tomato).

Problem 4: Skolemize sentences A,B, and C above.

Answer:

A.1 $\neg$stock$(S, P) \vee$ inst(sk0(S, P),P).

A.2 $\neg$stock$(S, P) \vee$ at(sk0(S, P),S).

A.3 $\neg$inst$(I, P) \vee \neg$at$(I, S) \vee$ stock(S, P).

B. $\neg$stock$(S, P) \vee$ carry(S, P).

C. $\neg$carry(deli94,P) $\vee$ carry(superxxx,P).