

# Logica e Algebra

24 luglio 2014 - Laboratorio

Si considerino le seguenti affermazioni:

“Alcuni filosofi ammirano Aldo. Tutte le persone sagge ammirano i buoni filosofi, e solo chi non è saggio ammira un cattivo filosofo. I buoni filosofi sono saggi. Aldo non ammira alcun filosofo.  
Dunque non tutti i filosofi sono buoni filosofi.”

Utilizzando un opportuno linguaggio si traducano le affermazioni precedenti in formule della logica del primo ordine e si formalizzi il problema nella sintassi di SPASS.

## Soluzione

Per formalizzare le affermazioni del problema è necessario aggiungere al linguaggio dei simboli extralogici (ovvero costanti, lettere funzionali o lettere predicative) per esprimere gli eventuali nomi, funzioni e relazioni che compaiono. Nel nostro caso dobbiamo sicuramente introdurre:

- una costante “a” da interpretarsi come “Aldo”;
- un simbolo predicativo unario “F” per il quale la formula atomica  $F(x)$  si interpreta come “ $x$  è un filosofo”;
- un simbolo predicativo unario “BF” per il quale la formula atomica  $BF(x)$  si interpreta come “ $x$  è un buon filosofo”;
- un simbolo predicativo unario “S” per il quale la formula atomica  $S(x)$  si interpreta come “ $x$  è saggio”;
- un simbolo predicativo binario “AM” per il quale la formula atomica  $A(x, y)$  si interpreta come “ $x$  ammira  $y$ ”.

Non è invece necessario aggiungere una lettera predicativa specifica per interpretare la proprietà “essere un cattivo filosofo”, visto che può essere reso semplicemente con la negazione di “essere un buon filosofo” (analogamente non serve una lettera per esprimere “non essere saggio”).

Osserviamo poi che, benché non esplicitamente espresso, bisogna formalizzare la relazione che intercorre tra “essere un filosofo” e “essere un buon filosofo”, ovvero che ogni buon filosofo è un filosofo, in formule

$$\forall x(BF(x) \Rightarrow F(x)).$$

Se invece si volesse far uso anche di una lettera predicativa  $CF$  per cui  $CF(x)$  si interpreta come “ $x$  è un cattivo filosofo”, allora è necessario aggiungere gli assiomi che esprimono che ogni filosofo è o buono o cattivo e non contemporaneamente entrambi (ovvero che gli insiemi dei buoni e cattivi filosofi determinano una partizione dell’insieme dei filosofi):

$$\begin{aligned}\forall x(F(x) \Rightarrow (BF(x) \vee CF(x))) \\ \neg \exists x(BF(x) \wedge CF(x))\end{aligned}$$

La frase “alcuni filosofi ammirano Aldo” si formalizza come

$$\exists x(F(x) \wedge AM(x, a)),$$

infatti la frase è equivalente ad affermare che esiste una persona che è un filosofo e ammira Aldo.

La frase “tutte le persone sagge ammirano i buoni filosofi” si formalizza come

$$\forall x \forall y((S(x) \wedge BF(y)) \Rightarrow AM(x, y)),$$

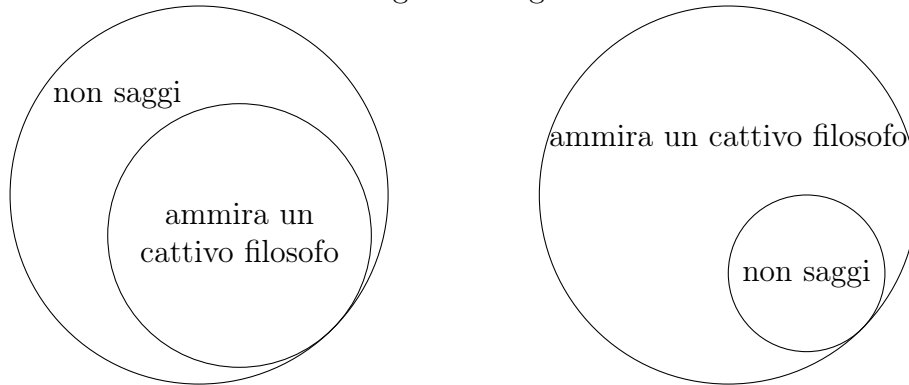
infatti la frase è equivalente ad affermare che presi comunque due individui, se uno saggio e l’altro è un buon filosofo, allora sicuramente il primo deve ammirare il

secondo.

La frase “solo chi non è saggio ammira un cattivo filosofo” si formalizza come

$$\forall x((\exists y(\neg BF(y) \wedge AM(x, y))) \Rightarrow \neg S(x)),$$

infatti la frase è equivalente ad affermare che l’insieme delle persone che ammirano Aldo è contenuto in quello delle persone non sagge. Se la cosa non ci convince, proviamo a considerare i due seguenti diagrammi di Eulero-Venn:



In quello di destra osserviamo che ci sono delle persone sagge (ovvero nel complementare dell’insieme dei non saggi) che ammirano cattivi filosofi, dunque in tale situazioni non solo i non saggi ammirano qualche cattivo filosofo, ma anche alcuni saggi. Dobbiamo dunque formalizzare la situazione a sinistra, ovvero che l’insieme degli ammiratori di cattivi filosofi è contenuto in quello delle persone non sagge.

La frase “i buoni filosofi sono saggi” si formalizza come

$$\forall x(BF(x) \Rightarrow S(x)).$$

La frase “Aldo non ammira alcun filosofo” si formalizza come

$$\forall x(F(x) \Rightarrow \neg AM(a, x)).$$

Infine la frase “non tutti i filosofi sono buoni filosofi” (che è la congettura del problema) si formalizza come

$$\exists x(F(x) \wedge \neg BF(x)).$$

In Spass il problema si formalizza così:

```
begin_problem(laboratorio24072014).

list_of_descriptions.
name(*laboratorio24072014*).
author(**).
status(unsatisfiable).
description(**).
end_of_list.

list_of_symbols.
  functions[(a,0)].
  predicates[(F,1),(BF,1),(S,1),(AM,2)].
end_of_list.

list_of_formulae(axioms).
formula(forall([x],implies(BF(x),F(x)))).
formula(exists([x],and(F(x),AM(x,a)))).
formula(forall([x,y],implies(and(S(x),BF(y)),AM(x,y)))).
formula(forall([x],implies(exists([y],and(AM(x,y),not(BF(y)))),not(S(x))))).
formula(forall([x],implies(F(x),not(AM(a,x))))).
formula(forall([x],implies(BF(x),S(x)))).
end_of_list.

list_of_formulae(conjectures).
formula(exists([x],and(F(x),not(BF(x))))).
end_of_list.

list_of_settings(SPASS).
{*
set_flag(DocProof,1).
*}
end_of_list.

end_problem.
```

che produce il seguente output:

```
-----SPASS-START-----
Input Problem:
1[0:Inp] || -> F(skc1)*.
2[0:Inp] || -> AM(skc1,a)*.
3[0:Inp] || BF(U) -> S(U)*.
4[0:Inp] || BF(U)* -> F(U).
5[0:Inp] || F(U) -> BF(U)*.
6[0:Inp] || AM(a,U)* F(U) -> .
7[0:Inp] || S(U) AM(U,V)* -> BF(V).
```

8[0:Inp] || BF(U) S(V) -> AM(V,U)\*.

This is a first-order Horn problem without equality.

This is a problem that has, if any, a finite domain model.

There are no function symbols.

This is a problem that contains sort information.

Axiom clauses: 7 Conjecture clauses: 1

Inferences: IEmS=1 ISoR=1 IORe=1

Reductions: RFMRR=1 RBMRR=1 RObv=1 RUnC=1 RTaut=1

RSST=1 RSSi=1 RFSub=1 RSub=1 RCon=1

Extras : Input Saturation, Always Selection, No Splitting, Full Reduction,  
Ratio: 5, FuncWeight: 1, VarWeight: 1

Precedence: div > id > AM > S > BF > F > a > skc0 > skc1

Ordering : KBO

Processed Problem:

Worked Off Clauses:

Usable Clauses:

1[0:Inp] || -> F(skc1)\*.

9[0:Res:1.0,5.0] || -> BF(skc1)\*.

2[0:Inp] || -> AM(skc1,a)\*.

3[0:Inp] BF(U) || -> S(U)\*.

4[0:Inp] BF(U) || -> F(U)\*.

5[0:Inp] F(U) || -> BF(U)\*.

6[0:Inp] F(U) || AM(a,U)\* -> .

8[0:Inp] S(U) BF(V) || -> AM(U,V)\*.

7[0:Inp] S(U) || AM(U,V)\* -> BF(V).

Given clause: 1[0:Inp] || -> F(skc1)\*.

Given clause: 9[0:Res:1.0,5.0] || -> BF(skc1)\*.

Given clause: 2[0:Inp] || -> AM(skc1,a)\*.

Given clause: 3[0:Inp] BF(U) || -> S(U)\*.

Given clause: 4[0:Inp] BF(U) || -> F(U)\*.

Given clause: 5[0:Inp] F(U) || -> BF(U)\*.

Given clause: 6[0:Inp] F(U) || AM(a,U)\*+ -> .

Given clause: 8[0:Inp] S(U) BF(V) || -> AM(U,V)\*.

Given clause: 17[0:SSi:16.1,5.1] S(a) F(U) || -> .

Given clause: 7[0:Inp] S(U) || AM(U,V)\*+ -> BF(V).

SPASS V 3.0

SPASS beiseite: Proof found.

Problem: /tmp/webpass-webform\_2014-07-25\_12:01:37\_118861.txt

SPASS derived 6 clauses, backtracked 0 clauses and kept 13 clauses.

SPASS allocated 587 KBytes.

SPASS spent 0:00:00.01 on the problem.

0:00:00.00 for the input.

0:00:00.00 for the FLOTTER CNF translation.

0:00:00.00 for inferences.

0:00:00.00 for the backtracking.

0:00:00.00 for the reduction.

Here is a proof with depth 2, length 15 :

```
1[0:Inp] || -> F(skc1)*.
2[0:Inp] || -> AM(skc1,a)*.
3[0:Inp] BF(U) || -> S(U)*.
5[0:Inp] F(U) || -> BF(U)*.
6[0:Inp] F(U) || AM(a,U)*+ -> .
7[0:Inp] S(U) || AM(U,V)*+ -> BF(V).
8[0:Inp] S(U) BF(V) || -> AM(U,V)*.
9[0:Res:1.0,5.0] || -> BF(skc1)*.
16[0:Res:8.2,6.1] S(a) BF(U) F(U) || -> .
17[0:SSi:16.1,5.1] S(a) F(U) || -> .
18[0:SoR:17.0,3.1] F(U) BF(a) || -> .
19[0:Res:2.0,7.1] S(skc1) || -> BF(a)*.
21[0:SSi:19.0,1.1,9.0,3.0] || -> BF(a)*.
22[0:MRR:18.1,21.0] F(U) || -> .
23[0:UnC:22.0,1.0] || -> .
```

Formulae used in the proof : axiom1 axiom5 conjecture0 axiom4 axiom3 axiom2

-----SPASS-STOP-----

che prova la correttezza della congettura.