

## Tutorial 1: First-order logic in database

(1) Consider the following database DB of a fictional company with three relations:

EMPL			MANAGING		JUNIORS
1	Alice	software	1	5	9
2	Bob	network	1	8	8
3	Chen	hardware	6	4	5
4	David	software	1	9	11
5	Ed	admin	7	6	2
6	Fan	network	3	2	6
7	Gil	hardware	11	4	4
8	Huo	network	7	10	10
9	Imri	software	1	8	
10	Jay	admin	8	7	
11	Kuo	software	10	3	

- A tuple  $\text{EMPL}(x, y, z)$  means an *employee* with id  $x$ , name  $y$ , and department  $z$ .
- A tuple  $\text{MANAGING}(x, y)$  means employee with id  $x$  is *managing* employee with id  $y$ .
- A tuple  $\text{JUNIORS}(x)$  means employee with id  $x$  is still a *junior* in the company.

Evaluate the following queries.

$$\begin{aligned}
 \varphi_1(y) &:= \exists x \exists z \text{EMPL}(x, y, z) \\
 \varphi_2(x_1, y_1, z_1, x_2, y_2, z_2) &:= \text{EMPL}(x_1, y_1, z_1) \wedge \text{EMPL}(x_2, y_2, z_2) \wedge \text{MANAGING}(x_1, x_2) \\
 \varphi_3(y) &:= \exists x \exists z \exists w (\text{EMPL}(x, y, z) \wedge \text{MANAGING}(x, w)) \\
 \varphi_4(y) &:= \exists x \exists z (\text{EMPL}(x, y, z) \wedge \neg(\exists w \text{MANAGING}(x, w))) \\
 \varphi_5(y) &:= \exists x \exists z (\text{EMPL}(x, y, z) \wedge \exists w (\neg \text{MANAGING}(x, w)))
 \end{aligned}$$

(2) Write down the FO queries that result in the following.

- The names of all the junior employees.
- The names of the employees that are managed by somebody.
- The names of the employees that are *not* managed by anybody.
- The names of the employees that manage somebody in the *same* department.
- The names of the employees that manage somebody in *different* departments.

(3) Let  $\text{DB} = (U, R, S, T)$  be a database with  $U$  being the domain, and  $R, S, T$  are relation names with arity 2, 2, 1, respectively. We assume that  $U, R, S, T$  are given as the list data structures (like in Java or C, for example), that comes with a method to check whether it contains a certain element.

Write the algorithms to evaluate the results of the following queries on database DB. You can assume that the output of each query is to be stored in a list too.

$$\begin{aligned}\alpha_1(x, y) &:= R(x, y) \wedge \neg S(x, y) \\ \alpha_2(x) &:= \exists w (S(x, w) \wedge T(w)) \\ \alpha_3(x) &:= \exists z \exists y (R(x, z) \wedge S(z, y)) \\ \alpha_4(x) &:= \neg \exists z \exists y (R(x, z) \wedge S(z, y)) \\ \alpha_5(x) &:= \exists z \exists y (R(x, z) \wedge S(z, y)) \wedge \neg \exists w (S(x, w) \wedge T(w)).\end{aligned}$$