

**Computer Science CSC 324S: Programming Languages**  
**University of Toronto – St. George Campus**  
**Assignment PrologA**  
**Due Date: Tuesday, 23 March, 6:10pm exactly**

**1. Classical First-Order Logic**

This question uses the following predicates:

- person*( $x$ ), which means that  $x$  is a person,
- male*( $x$ ), which means that person  $x$  is male,
- female*( $x$ ), which means that person  $x$  is female,
- mother*( $x, y$ ), which means that  $y$  is the mother of  $x$ ,
- father*( $x, y$ ), which means that  $y$  is the father of  $x$ ,
- married*( $x, y$ ), which means that  $x$  and  $y$  are married,
- parent*( $x, y$ ), which means that  $y$  is a parent of  $x$ ,
- brother*( $x, y$ ), which means that  $x$  is a brother of  $y$ ,
- sister*( $x, y$ ), which means that  $x$  is a sister of  $y$ ,
- son*( $x, y$ ), which means that  $x$  is a son of  $y$ ,
- daughter*( $x, y$ ), which means that  $x$  is a daughter of  $y$ ,
- name*( $x, y$ ), which means that  $x$ 's name is  $y$ ,
- badbreath*( $x$ ), which means that  $x$  has bad breath.
- friend*( $x, y$ ), which means that  $x$  is the friend of  $y$ .

Further, we define the constants **paul**, **petra**, **matt**, **sally**, **susan**, **i**.

Using only these predicates and constants, translate each of the English statements below into classical first-order logic. Do not use any function symbols, additional constant symbols or arithmetic symbols. You may use the equality predicate if necessary.

1. There is at least one person who has a sibling (brother or sister).
2. Some person's sister has bad breath.
3. Matt has a friend who is married to Petra.
4. Someone's children are called Paul and Susan.
5. Matt and Sally are not parents.
6. There is a person with bad breath who is not married to anyone.
7. Every mother's son with bad breath does not have a sister who is married.
8. Someone's son's name is Matt, who is married to Sally.
9. People who are siblings are not both named Matt.
10. Every person has two parents and four grandparents, all distinct.

2.

For each of the following pairs, state whether or not the atomic formulae unify. In each case, state what the most general unifier (mgu) is, or show that unification is impossible. In this problem,  $U, V, X, Y, Z$  are variables, and  $a, b, c, d$  are constant symbols.

- $s(X, d)$  and  $t(Z, Z)$
- $t(a, Y, d)$  and  $t(X, X, Z)$
- $q(a, W, c)$  and  $q(X, b, X)$
- $p(V, a, f(V))$  and  $p(c, X, W)$
- $s(f(a(W)), X, X)$  and  $s(X, X, Z)$
- $t(f(a), X, f(X))$  and  $t(Z, g(a), f(Z))$
- $q(U, f(a, U), X)$  and  $q(X, X, Z)$
- $r(X, h(X), f(d))$  and  $r(g(a), h(Z), Z)$

Note: not every unifier is most general! A mark will be given only for the most general unifier.

3.

Write each of the following statements of first-order logic as a set of Horn clauses, or say why the conversion cannot be made. Use logical equivalences like the ones on slide 230 of the slides from Chechik's slides from last term, linked from the course web page. (Look for "notes.html", and use the Lecture 17 slides.) Don't do Skolemization, and don't forget that

$$\neg(\forall X f(x)) = \exists X \neg f(x)$$

$$\neg(\exists X f(x)) = \forall X \neg f(x)$$

1.  $\forall X \exists Y [\neg a(X, Y) \rightarrow b(X, Y)]$
2.  $\forall X \exists Y [a(X, Y) \rightarrow \neg b(X, Y)]$
3.  $\forall X, Y [a(X, Y) \leftarrow \neg \exists Z [b(Z, X) \rightarrow c(X, Y)]]$
4.  $\forall X [f(X) \rightarrow \forall Y, Z [g(X, Z) \wedge \exists W [a(Z, W) \wedge k(Y, W)]]]$
5.  $\forall Y [f(Y) \leftarrow \exists X \forall Z [a(X, Z) \leftarrow b(Y, Z)]]$
6.  $\forall X \exists Y [c(X, Y) \rightarrow (f(X) \vee \forall Y, Z [g(X, Z) \wedge \neg(\forall W \exists X [a(Z, W) \wedge k(Y, W)])])]$