Problems

These problems will be discussed in week 6.

---

1 Use the unification algorithm to unify `append(cons(1, empty), cons(2, cons(3, empty)), L)` with `append(cons(X, Xs), Ys, cons(X, Zs))`. Indicate the arguments to `unify` at each recursive call and the value of the substitution σ after each call.

2 Use the resolution algorithm to find a proof of `male(oskar)` using the initial example in the lecture notes on Prolog.

3 Extend **While** with a `halt` statement that terminates the execution of the program in the current state. Define $\mathcal{S}_{cs}[\![\texttt{halt}]\!]$ using continuation style semantics.

4 (Ex 6.13) Assume that $\mathcal{S}'_{cs}[\![S_i]\!]c\ \sigma = c(\mathcal{S}_{ds}[\![S_i]\!]\sigma)$ for $i = 1, 2$. Show that $\mathcal{S}'_{cs}[\![S]\!]c\ \sigma = c(\mathcal{S}_{ds}[\![S]\!]\sigma)$ when $S =$ `if b then` S_1 `else` S_2.

5 Show that `s+x*y=`p is an invariant for

```
while ¬(x=0) do (s:=s+y; x=x-1)
```

6 Provide an axiomatic inference rule for `repeat c until b` that does not rely on the `while` statement.

7 Find an appropriate invariant to use in the `while`-rule for proving

$\{\texttt{y} = n\}$ `while ¬(y=0) do (y := y-1; x := 2*x)` $\{\texttt{x} = 2^n\}$.

Hint. The invariant should be an equation relating `x, y` and 2^n.

8 Show: If $P \Rightarrow Q$ then $\mathcal{S}_{\text{wp}}[\![S]\!]P \Rightarrow \mathcal{S}_{\text{wp}}[\![S]\!]Q$ for all S in Stm.