Vlad Bogdan-Tudor, 917
VBT

$$S = \{ P(z) \vee R(z),\ \neg P(x),\ \neg R(x) \vee \neg Q(x),\ W(g(a)),$$
$$Q(y) \vee \neg W(y) \}$$

Lock resolution: We index the clauses arbitrarily and we only resolve the ones that can be resolved if the indexes are smallest.

- 1st indexing:

$C_1 = {}_{(1)}P(z) \vee_{(2)} R(z)$

$C_2 = {}_{(3)}\neg P(x)$

$C_3 = {}_{(4)}\neg R(x) \vee_{(5)} \neg Q(x)$

$C_4 = {}_{(5)}W(g(a))$

$C_5 = {}_{(6)}Q(y) \vee_{(7)} \neg W(y)$

Vlad Bogdan-Tudor, 917, VldPd

We know that lock resolution is sound so if we derive the empty clause from a set S of propositional/predicate clauses ⇒ S is inconsistent.

Since we are working in predicate logic, we will use the concept of substitutions.

A substitution is a mapping from a set of variables into a set of terms. We denote it by $\Theta = [x_1 \leftarrow t_1, \ldots, x_k \leftarrow t_k]$ where x_i can only be variables and t_i are terms. Also x_i must not be a subterm of t_i.

Vlad Bogdan-Tudor, 917, VLBol

$C_6 = \text{Res}^{\text{lock}}_{\theta_1 = [z \leftarrow x]}(C_1, C_2) =_{(2)} r(x)$

$C_7 = \text{Res}^{\text{lock}}(C_6, C_3) =_{(5)} \neg q(x)$

$C_8 = \text{Res}^{\text{lock}}_{\theta_2 = [x \leftarrow y]}(C_5, C_7) =_{(7)} \neg w(y)$

$C_9 = \text{Res}^{\text{lock}}_{\theta_3 = [y \leftarrow g(a)]}(C_8, C_4) = \square$

We derived the empty clause from $S \Rightarrow S$ is inconsistent.

2nd indexing:

$C_1 =_{(6)} p(z) \vee_{(5)} r(z)$

$C_2 =_{(8)} \neg p(x)$

$C_3 =_{(2)} \neg r(x) \vee_{(1)} \neg q(x)$

$C_4 =_{(7)} w(g(a))$

$C_5 =_{(3)} q(y) \vee_{(4)} \neg w(y)$

$C_6 = \text{Res}^{\text{lock}}_{[\theta_1 = y \leftarrow x]}(C_3, C_5) =_{(2)} \neg r(x) \vee_{(4)} \neg w(x)$

$C_7 = \text{Res}^{\text{lock}}_{[\theta_2 = z \leftarrow x]}(C_6, C_1) =_{(4)} \neg w(x) \vee_{(6)} p(x)$

$C_8 = \text{Res}^{\text{lock}}_{[\theta_3 = x \leftarrow g(a)]}(C_7, C_4) =_{(6)} p(g(a))$

Vlad Bogdan-Tudor, 917
VlaBd

$$C_9 = Res^{lock}_{[\Theta_4 = x \leftarrow g(4)]}(C_2, C_8) = \square$$

We derived the empty clause from $S \Rightarrow S$ is inconsistent.