Unification
Unification

Two terms $s$ and $t$ are unifiable iff there exists a substitution (called unifier) s.t. $\theta(s) = \theta(t)$.

Example

- $f(x, g(x, a))$ and $f(f(a), y)$ are unifiable with $\theta = \{x/f(a), y/g(f(a), a)\}$

\[
\begin{align*}
  f(x, g(x, a)) & \equiv f(f(a), y) \\
  f(f(a), g(f(a), a)) & = f(f(a), g(f(a), a))
\end{align*}
\]

- $f(x, g(x, a))$ and $f(f(a), f(b, a))$ are not unifiable.
Principal substitutions

Let $\theta$ and $\tau$ be two substitutions and $S$ be a set of substitutions.

- The composition of $\theta$ and $\tau$ is $(\theta \circ \tau)(x) = \theta(\tau(x))$ for every variable $x \in X$.
- $\theta$ is an instance of $\tau$ (or $\tau$ is more general than $\theta$) iff there exists a substitution $\rho$ s.t. for every variable $x \in X$, $(\rho \circ \tau)(x) = \theta(x)$.
- $\tau \in S$ is principal (or most general) iff every substitution $\theta \in S$ is an instance of $\tau$.

Example

Let $\sigma_1 = \{y/b, z/h(c)\}$ and $\sigma_2 = \{x/f(y), y/z\}$.

$\sigma_1 \circ \sigma_2 = \{x/f(b), y/h(c), z/h(c)\}$.

$\sigma_2$ is more general than $\sigma_1 \circ \sigma_2$. 
Theorem

Let $S$ be a non-empty set of unifiers of $s$ and $t$. Then, there exists a principal unifier $\theta \in S$ s.t. for every $\tau \in S$, $\theta$ is more general than $\tau$. Moreover, this principal unifier is unique modulo renaming.
A substitution $\theta$ is idempotent iff $\theta \circ \theta = \theta$.

Example

\{y/b, z/h(c)\} is idempotent.
\{x/f(y), y/z\} is not idempotent.

Theorem

If $s$ and $t$ are unifiable, then there exists a principal unifier of $s$ and $t$ which is idempotent.

How we can construct this unifier?
An *equational system* is a set of *equations* of the form $s \doteq t$. 
An equational system $E$ is **unifiable** iff there exists a unifier (called **solution**) for all the equations of $E$.

**Finite** equational systems are denoted $\{s_1 \doteq t_1, \ldots, s_n \doteq t_n\}$. 
Solved forms

The equational system $E = \{s_1 \doteq t_1, \ldots, s_n \doteq t_n\}$ is in solved form iff

- All the $s_i$ are distinct variables.
- No $s_i$ appears in $t_j$.

Example

$E_0 = \{x \doteq y, z \doteq f(a)\}$ is in solved form but $E_1 = \{x \doteq y, x \doteq f(a)\}$,
$E_2 = \{x \doteq y, y \doteq f(a)\}$,
$E_3 = \{x \doteq z, y \doteq f(y)\}$ do not.

Notation: For the solved system $E = \{x_1 \doteq t_1, \ldots, x_n \doteq t_n\}$ we note $\vec{E}$ the substitution $\{x_1/t_1, \ldots, x_n/t_n\}$. 

The transformation rules

\[
\begin{align*}
E \cup \{ s \doteq s \} & \quad \text{(erase)} \\
\hline
E & \\
\end{align*}
\]

\[
\begin{align*}
E \cup \{ t \doteq x \} & \quad t \notin \mathcal{X} \quad \text{(orient)} \\
\hline
E \cup \{ x \doteq t \} & \\
\end{align*}
\]

\[
\begin{align*}
E \cup \{ f(s_1, \ldots, s_n) \doteq f(t_1, \ldots, t_n) \} & \quad \text{(decompose)} \\
\hline
E \cup \{ s_1 \doteq t_1, \ldots, s_n \doteq t_n \} & \\
\end{align*}
\]

\[
\begin{align*}
E \cup \{ x \doteq s \} & \quad x \in \text{Var}(E) \quad x \notin \text{Var}(s) \quad \text{(substitute)} \\
\hline
E\{x\doteq s\} \cup \{ x \doteq s \} & \\
\end{align*}
\]
The unification algorithm

1. Take an equational system $E$
2. Compute a new system $P$ by applying the transformation rules as far as possible.
3. If the system $P$ is in solved form
   - then send the answer $\vec{P}$
   - else fail
Example

Unification of the system \( \{p(a, x, f(g(y))) \doteq p(z, f(z), f(u))\} \)

\[
\begin{align*}
p(a, x, f(g(y))) & \doteq p(z, f(z), f(u)) \\
a \doteq z, x \doteq f(z), f(g(y)) & \doteq f(u) \\
z \doteq a, x \doteq f(z), f(g(y)) & \doteq f(u) \\
z \doteq a, x \doteq f(a), f(g(y)) & \doteq f(u) \\
& \quad \rightarrow z \doteq a, x \doteq f(a), g(y) \doteq u \\
& \quad \rightarrow z \doteq a, x \doteq f(a), u \doteq g(y) (\text{solved form})
\end{align*}
\]

yields the (idempotent) substitution \( \{z/a, x/f(a), u/g(y)\} \).
Soundness and completeness of the algorithm

Theorem

*The algorithm terminates.*

Theorem

*(Soundness)* If the algorithm finds a substitution \( \vec{S} \) for the problem \( P \), then \( P \) is unifiable and \( \vec{S} \) is a m.g.u. of \( P \).

That is,

If \( P \) is not unifiable, then the algorithm fails.

Theorem

*(Completeness)* If the system \( P \) is unifiable, then the algorithm computes the m.g.u. of \( P \).

That is,

If the algorithm fails, then the system \( P \) is not unifiable.