

<b>Date:</b> <b>Ex No:</b>  7.1	<b>Title of the Lab</b> Implementation of Unification in SWI Prolog	<b>Name:</b> Mainak Chaudhuri <b>Registration Number:</b>  RA1911027010039 <b>Section:</b> N1 <b>Lab Batch:</b> 1 <b>Day Order:</b> 3
--	---	---

AIM:

To implement Unification in SWI Prolog.

Description of the Concept or Problem given:

Prolog uses the unification technique, and it is a very general form of matching technique. In unification, one or more variables being given value to make the two call terms identical. This process is called binding the variables to values. For example, Prolog can unify the terms `cat(A)`, and `cat(mary)` by binding variable `A` to atom `mary` that means we are giving the value `mary` to variable `A`.

Manual Solution:

1. If `Y1` or `Y2` is a variable or constant, then:
  - a) If `Y1` , or `Y2` are identical, then return `NIL`.
  - b) Else if `Y1` is a variable,
    - a. then if `Y1`, occurs in `Y2`, then return `FAILURE`
    - b. Else return `{{Y2,/Y1}}`.
  - c) Else if `Y2` is a variable,
    - a. If `Y2` occurs in `Y1`, then return `FAILURE`,
    - b. Else return `{{Y1/Y2}}`.
  - d) Else return `FAILURE`.
2. If the initial Predicate symbol in `Y1`, and `Y2` are not same, then return `FAILURE`.
3. If `Y1` and `Y2` have a different number of arguments, then return `FAILURE`.
4. Set Substitution `set(SUBST)` to `NIL`.
5. For `i=1` to the number of elements in `Y1`.
  - a) Call Unify function with the `ith` element of `Y1`, and `ith` element of `Y2`, and put the result into `S`.
  - b) If `S=failure` then returns `Failure`
  - c) If `S != NIL` then do,
    - a. Apply `S` to the remainder of both `L1` and `L2`.
    - b. `SUBST = APPEND(S, SUBST)`.
6. Return `SUBST`.

Screenshots of the Outputs:

## 18CSC305J Artificial Intelligence Lab

```
SWI-Prolog (AMD64, Multi-threaded, version 8.4.2)
File Edit Settings Run Debug Help
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.2)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?- employees(_,name(sid),_).
true.

?- employees(X,name(sid),Y).
X = 102
Y = address(mexico).

?- employees(101,name(B),C).
B = adi
C = address(ny)
Unknown action: 0 (h for help)
Action? .

?- employees(A.name(B),C).
A = 100.
B = yuvraj.
C = address(canada) .

?- employees(101,Name,Address).
Name = name(adi).
Address = address(ny).

?- employees(ID,Name,Address).
ID = 100.
Name = name(yuvraj).
Address = address(canada)
```

```
employee.pl
File Edit Browse Compile Prolog Pcs Help
employee.pl
employees(100, name(yuvraj), address(canada)).
employees(101, name(adi), address(ny)).
employees(102, name(sid), address(mexico)).
employees(103, name(mayank), address(la)).
employees(104, name(shivam), address(nc)).

c:/users/admin/onedrive/desktop/lab 7/employee.pl compiled
Line: 5
```

Signature of the Student

[MAINAK CHAUDHURI]