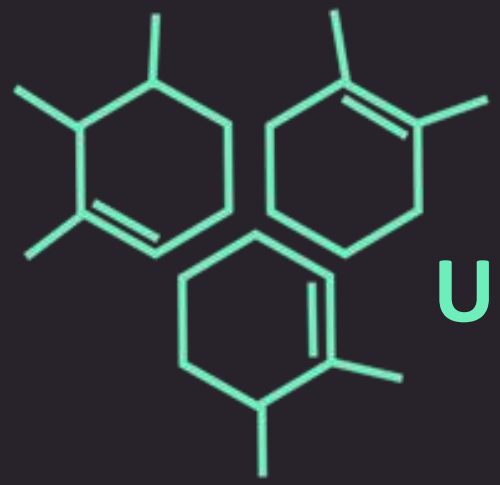
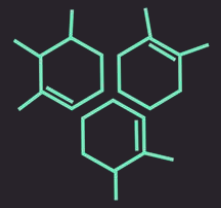
SUHAS P
1GA19CS159

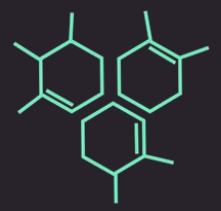


Unification Algorithm



Unification Algorithm

- In propositional logic it is easy to determine that two literals cannot be true at the same time
- Simply look for L and ~L . In predicate logic, this matching process is more complicated, since bindings of variables must be considered
- In order to determine contradictions we need a matching procedure that compares two literals and discovers whether there exist a set of substitutions that makes them identical
- The process of finding a substitution for predicate parameters is called **unification**
- We need to know:
  - that 2 literals can be matched
  - the substitution is that makes the literals identical



Unification Algorithm

unify(L1,L2)

1) If L1 and L2 is a variable or constant, then
    a) If L1 and L2 are identical, return NIL
    b) Else if L1 is a variable, then if L1 occurs in L2 then return FAIL, else return {(L1/L2}
    c) Same as (b) for L2
    d) Else return FAIL
2) If the initial predicate symbols in L1 and L2 are not identical, then return FAIL
3) If L1 and L2 have different number of arguments, then FAIL
4) SET SUBST to NIL
4) Return SUBST

THANK YOU