Things to remember

# Logically equivalent

Two formulas A, B are said to be logically equivalent iff A ↔ B is a tautology. Iff we have V(A) = V(B) for every assignment V. **Same truth table**

A → B is logically equivalent to ¬ A v B

# Equisatisfiable

Two formulas A, B are said to be equisatisfiable iff A is satisfiable iff B is satisfiable

# NNF (Negation Normal Form)

No implications and all negations are in front of atoms

((¬ P ^ (Q v R) ) v S) is NNF but not CNF

( ¬ P ^ Q) **v** ( ¬ P)

# CNF (Conjunction Normal Form)

It's a conjunction of clauses

(PvQ) **^** (Pv¬ Q)

# Horn Clauses

It's a clauses with at most one positive literal

¬ P v ¬ Q is a Horn Clause

¬ P v Q v ¬ R is a Horn Clause

P v Q v R is NOT a Horn Clause

# Subsumption and Unit resolution

| **Subsumption** | **Unit Resolution** |
| --- | --- |
| Remove clauses | Remove literals |
| P v Q v ¬R if V(P)=1 than I can remove the clause | P v Q v ¬R if V(P)=0 than I can remove the literal P, it becomes Q v ¬R |
| If you remove all the clause you get the empty set of clauses | If you remove all the literals you get the empty clause |
| SAT | Failed branch, get ⬜️ apply backjumping |

# Herbrand Universe

If the herbrand universe is finite, we have finite formulas, we can check all of them

If the herbrand universe is infinite, we get infinitely many formulas, feed the SAT-solver a finite number until we discover the inconsistency. There is the risk of no termination

Skolemization transform non-universal formulas in universal sentences, keeping satisfiability

# Prenex Normal Form

Qx1, Qx2, Qx3 M

Q: quantifiers (or ∃ or ∀ )

M quantifiers free, called matrix

!!! if you change the order of quantifiers you change the meaning !!!

Transformation

| a ^ ∀xb | ∀ x (a^b) |
| --- | --- |
| ∀xB ^ A | ∀x(B ^ A) |
| ∃xB^A | ∃x(B^A) |
| A v ∀xB | ∀x(A v B) |
| ∀xB v A | ∀x(B v A) |
| A v ∃xB | ∃x(A v B) |
| ∃xB v A | ∃x(B v A) |
| ¬∃x A | ∀x ¬ A |
| ¬∀x A | ∃x ¬ A |
| **A → ∃xB** | **∀x(A→B)** |
| **A → ∀xB** | **∃x(A→B)** |
| **∃xB → A** | **∀x(B→A)** |
| **∀xB → A** | **∃x(B→A)** |
| ∀xB ^ ∀xA | ∀x(B^A) |

!!! you can always rename !!!

P(x) ^ ∀xR(x,y) ∀ x1 (P(x) ^ R(x1,y))

Change the name of all bound variable in such a way there is no free and bound variable

# Z3 tricks

*A iff B is* A ↔ B is (A → B) ^ (B → A) this is equal to “e viceversa”

*if then else*  is (ite <condition> value1 value2)

# Should I use → or ^ ?

*all humans are animals*

∀x(H(x) ^ A(x)) and ∀x(H(x) → A(x)) have different meaning, using ^ means that **all** elements are humans and animals, so we should use an implies, that is like an if

# Types of exercices

1. Prove something (pinocchio, knights and knives). You have to negate the thesis, assert the negation of the thesis and get an unsat of the answer
   1. if it UNSAT, than is true
   2. if is SAT, than it is false
2. Find something (color countries), don’t need to negate anything

# Steps for syllogism

1. take the English sentence
2. translate to mathematical logic
3. use skolna constant to remove existential quantifiers ( ∃)
4. get an universal problem with a finite herbrand universe
5. make all instantiations
6. apply a prepositional method (resolution, DPLL) to see if the set is consistent or not
7. If I get empty clause (⬜): syllogism is valid, if I don’t get empty clause then the syllogism is not valid