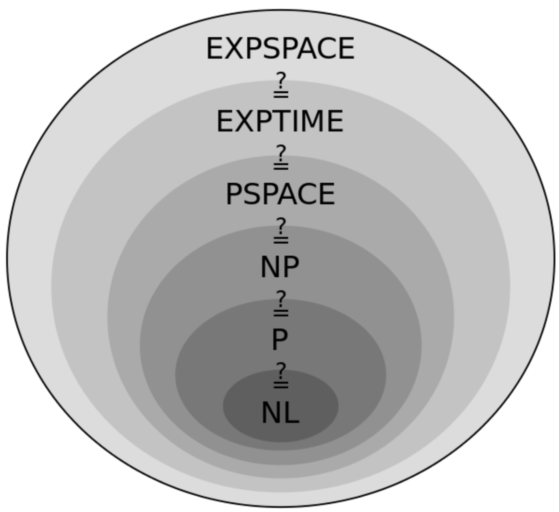
CS 511 Formal Methods, Fall 2018	Instructor: Assaf Kfoury
Lecture 8: QBF	
Oct 2, 2018	Jiayu Zhang

(These lecture notes are **not** proofread and proof-checked by the instructor.)

First the instructor introduced QBF. We need to know that QBF is not first-order logic(predicate logic).

In QBF we need to pay attention to the binding occurrences. When we do substitution, it's easy to make mistake here.

Deciding the validity of QBF is a PSPACE-complete problem. Recall that satisfiability problem of propositional WFF is NP-complete. The relation of different complexity can be illustrated as follows:



(This picture is from https://en.wikipedia.org/wiki/PSPACE)

We can solve all the PSPACE problem if we have a PSPACE-complete problem solver as an oracle.

Then the instructor gave an example for open and closed QBF:

$$\psi = \forall x \forall y (x \to y) \leftrightarrow (\neg x \vee y)$$

$$\varphi = (x \to y) \leftrightarrow (\neg x \vee y)$$

In φ, x, y are free variables. φ is open. While in ψ, there is no free variable, and ψ is closed.

There is no truth table for QBF. For formal semantics, see lecture notes. (Inductive definition)

Satisfiable means for one assignment, the formula is true. While valid means for all cases, it's true.

For formula transformation, there is a related concept: code reuse. In practice we hope the code reuse is safe. Similarly, in QBF, we can transform $\neg\varphi \wedge \neg\psi$ to $\neg(\varphi \wedge \psi)$, then we reuse $\neg$.

Then the instructor gives an example for transforming QBF.

$$(\forall x \varphi) \to \psi \equiv \neg(\forall x \varphi) \vee \psi$$

$$(\exists x \neg\varphi) \vee \psi \equiv \exists x(\neg\varphi \vee \psi) \text{ (Provided x is not free)}$$

And these two transformations can be visualized with parse tree.(See lecture notes)