

Airport example continued:

A heap is a semicomplete binary tree with  $n$  nodes so its height is  $\lfloor \lg n \rfloor$ .

Operations	Heap
Op <sub>1</sub>	$\Theta(\lg n)$
Op <sub>2</sub>	$\Theta(1)$
Op <sub>3</sub>	$\Theta(\lg n)$

To remove the top of a heap, we remove the top node and place the last element at the top of the heap. We then fix (heapify) the heap by comparing the top to its 2 children and switching nodes so that the greatest of the three nodes is on top, and then recursing downward. This procedure takes no more than the height of the tree, or  $\lg n$ , steps.

NP-Completeness:

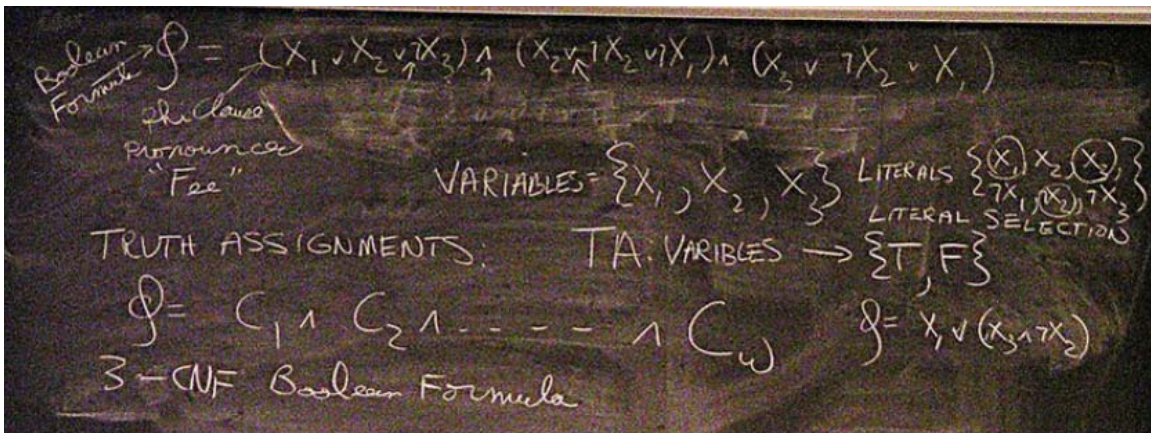
3-SAT:

A Boolean formula  $\Phi = (x_1 \vee x_2 \vee \neg x_3) \wedge (x_2 \vee \neg x_2 \vee \neg x_1) \wedge (x_3 \vee \neg x_2 \vee x_1)$

This formula has 3 clauses ( $C_1, C_2$ , and  $C_3$ ).

The variables of  $\Phi$  are  $\{x_1, x_2, x_3\}$ .

The literals of  $\Phi$  are  $\{x_1, \neg x_1, x_2, \neg x_2, x_3, \neg x_3\}$ .



A truth assignment maps the variables of  $\Phi$  to  $\{T, F\}$ , e.g.  $x_1 = T, x_2 = F, x_3 = F$ . With that assignment,  $\Phi$  evaluates to T. There are  $2^n$  possible truth assignments for  $n$  variables.

Truth table: a table of all possible truth assignments and  $\Phi$  values:

$X_1$	$X_2$	$X_3$	$\Phi$
T	T	T	
T	T	F	
T	F	T	T
T	F	F	
F	T	T	
F	T	F	
F	F	T	F
F	F	F	

If there exists a truth assignment such that  $\Phi = T$ , then  $\Phi$  is satisfiable.

The 3-SAT problem:

Input: A 3-CNF Boolean Formula  $\Phi$

Output: 1 if  $\Phi$  is satisfiable and 0 otherwise.

BF: brute force. Look through all possible truth assignments.

$T_{BF}^{WC}(n) = \Omega(2^n)$  (where  $n$  is the number of variables).

$$10^{10} \text{ computers} \times 10^{12} \frac{\text{operations}}{\text{second}} \times 26 \times 10^8 \frac{\text{seconds}}{\text{century}} \times 10^2 \text{ centuries} \approx 10^{34} \approx 2^{100} \text{ operations}$$

So if we used every operation on every man made machine ever performed, we'd be able to solve a 3-SAT problem, using the brute force method, with no more than roughly 100 variables. So exponential time is really slow, or "intractable. Polynomial time algorithms are good or "tractable."