Decision Tree

Lemma: Decision tree of height h has $L_h \le 2^h$ leaves

Proof by mathematical induction:

- $h = 1$: any tree of height 1 has $L_1 \le 2^1$ leaves
- $h-1 \to h$:
  - Let any tree of height $h - 1$ have $L_{h-1} \le 2^{h-1}$ leaves
  - Any tree of height h consists of a root and two subtrees of height at most $h - 1$
  - Therefore, $L_h = L_{h-1} + L_{h-1} \le 2^{h-1} + 2^{h-1} = 2^h$

Worst-Case Complexity of Sorting

- Theorem 2.32: The worst-case complexity of sorting n items by pairwise comparisons is $\Omega(n \log n)$
- Proof:
  - Any decision tree of height h has at most 2^h leaves (see Lemma, Slide 4)
  - The least height h such that $L_h = 2^h \geq n!$ leaves is

$$h \geq \log_2(n!) \cong n \log_2 n - 1.44\, n$$

Why *n!* ??

- There are at most 2^h leaves in a decision tree of height h, each corresponding to a different input order
- There are $n!$ permutations of the objects to be sorted, i.e. $n!$ different input (unsorted) orders
- Therefore, $2^h \geq n!$
  hence $h \geq \log_2(n!) \cong n \log_2 n - 1.44\, n$

- See http://knol.google.com/k/wlodzimierz-holsztynski/complexity-of-sorting/