**Problem 1.**

A binary tree T having L leaves must have node of depth at least.

If T is the decision tree for sorting algorithms, then sort n distinct values must be

That means, in the worst case, we need to at least comparisons.

When n = 4: comparisons.

**Problem 3.**

S = {80, 27, 72, 1, 27, 8, 64, 34, 16}

r[]:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 27 |  |  |  |  |  |  |  |  |
| 72 | 64 |  |  |  |  |  | 16 | 8 |
| 27 | 1 |  |  |  |  |  | 34 | 80 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |

q[]:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | 34 |  |  |  |  |  |
| 8 |  |  | 27 |  |  |  |  | 80 |
| 0 | 16 |  | 27 |  |  |  | 64 | 72 |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |

Read q[] from left to right, bottom to top: 0, 8, 16, 27, 27, 34, 64, 72, 80