

Problem

Prove that $\text{TIME}(2^n) = \text{TIME}(2^{n+1})$.

Step-by-step solution

Step 1 of 1

Proof of $\text{TIME}(2^n) = \text{TIME}(2^{n+1})$

Classification of computation time can be done on the basis of the minimum time required by the most efficient algorithm to determine upper and lower bounds.

Usually Big-O notation is used to state the lower and upper bound which **hides smaller terms** and **constant factors**.

- Big-O notation is used to define the **time complexity** classes. So there is no effect of constant factors used in complexity calculation.

- Therefore the time complexity of the function 2^{n+1} in big-O notation is $O(2^n)$.

- Thus, $A \in \text{TIME}(2^n)$ if and only if $A \in \text{TIME}(2^{n+1})$.

- Hence, $\text{TIME}(2^n) = \text{TIME}(2^{n+1})$

[Comment](#)