Clusters.Top Students Association

This answer was provided by CS Place, click here to join the Discord.

# Question:

Suppose five algorithms $A_1 \ldots A_5$ solving the same problem with execution times, $t_1(n), \ldots, t_5(n)$, of the following asymptotic upper bounds. Put the algorithms in order regarding their execution speed from the fastest to the slowest!

i. $O(2^{n+2})$
ii. $O([n^n]^2)$
iii. $O(4^n)$
iv. $O(n^2!)$
v. $O(\log n^n)$
vi. $O(n^{[n^2]})$

# Answer:

o(logn$^n$ ) = o(n log n) is the fastest execution time

Below are the execution time from fast to slow

**o(logn$^n$ ) < o(2$^{n+2}$) < o( 4$^n$) < o( [n$^n$]$^2$) < o(n$^{[n\text{^}2]}$) < o(n$^2$!)**

That is o(logn$^n$ ) has the low execution time where as o(n$^2$!) has the high execution time..

Low execution time indicates that the algorithm is faster

High execution time indicates that the algorithm is slower..

Note:: algorithmic complexities from best to least is

*O(1), O(log n) , O(n) , O(n log n) , O(n$^2$), O(n$^3$), O(2$^n$), O(n!) These are the frequently seen complexities from best to least..*

*Tip:: If we want to check , other complexities which are good and which are not, simply substitute a integer value in the variable "n", and then calculate the time..*

*For example :: in the given case we have* o(2$^{n+2}$) , o( 4$^n$) , o( [n$^n$]$^2$) complexities ..

assume n =3 and substitute n value in each complexity

o(2$^{n+2}$) =o(2$^{3+2}$) =o( 2$^5$)=o(32)=32

o( 4$^n$)=o( 4$^3$) =o( 64)= 64

$o([n^n]^2) = o([3^3]^2) = o([27]^2) = o(729) = 729$

$o(2^{n+2})$ Has less execution time compared to $o([n^n]^2)$ . Lowest execution time is the fastest algorithm...