

Quelques équations courantes dans les recherches de complexité d'algorithmes récurrents, et leur solution

Equation	Solution
$T(n)=T(n-1)$	$T(n)=\Theta(1)$
$T(n)=T(n-1)+1$	$T(n)=\Theta(n)$
$T(n)=T(n-1)+n^k$	$T(n)=\Theta(n^{k+1})$
$a>1, T(n)=aT(n-1)+n^k$	$T(n)=\Theta(a^n)$
$a>1, T(n)=aT(n-1)+a^n+n^k$	$T(n)=\Theta(na^n)$
$a,b>1, T(n)=aT(n-1)+b^n$	$T(n)=\Theta(\text{Max}(a^n, b^n))$
$T(n)=2T(n/2)+1$	$T(n)=\Theta(n)$
$T(n)=2T(n/2)+n$	$T(n)=\Theta(n\log n)$
$T(n)=2T(n/2)+n^k$	$T(n)=\Theta(n^k)$
$k<\log_b a, T(n)=aT(n/b)+n^k$	$T(n)=\Theta(n^{\log_b a})$
$k=\log_b a, T(n)=aT(n/b)+n^k$	$T(n)=\Theta(\log n n^k)$
$k>\log_b a, T(n)=aT(n/b)+n^k$	$T(n)=\Theta(n^k)$