DS HW2

2018-13335 Kim Kyeong Hwan

September 24, 2020

1

3*$2^n \rightarrow n! \rightarrow 6*n^2 \rightarrow log_2 n \rightarrow log_2 log_2 n \rightarrow 2^{10}$

2

1)c
2)b
3)b
4)c
5)a

3

1)
$T(n) = 2T(n-1)+1$
$T(n)+1 = 2T(n-1)+2$
$T(n)+1 = 2(T(n-1)+1)(let T(n)+1 = T`(n))$
$T`(n) = 2T`(n-1)$
$T`(n) = 2^{n-1} * T`(1)$
$T(n) = 2^n - 1$

2)
Let T"(n) is Y machine time complexity
then $T"(n) = T(n)/128 = t/128$
$T"(n+7) = t$
Therefore We can process $n+7$ inputs in t seconds in Y

4

1)
This code runs until $s = 1 + 2 + ...i = i * (i + 1)/2 < n$ and it runs i times.
Therefore time complexity is $\Theta(\sqrt{n})$

2)
This code runs n times and m times.
So time complexity is $\Theta(n + m)$

3)
This code runs $n(n + 1)/2$ times and n times
So time complexity is $\Theta(n^2)$

4) Let running time $T(n)$, then $T(n) = T(n - 1) + c$ (c is constant)
$T(n) = T(n - 2) + 2c = ... = nc$
So time complexity is $\Theta(n)$

5

1)
Let running time $T(n)$
Then $T(n) = T(n - 1) + c$ (c is constant, $T(0) = 1$)
$T(n) = T(n - 1) + c = T(n - 2) + 2c = ... = T(0) + nc$
Therefore $T(n) = \Theta(n)$

2)
Let running time $T(n)$
Then $T(n) = T(n/2) + c$ (c is constant, $T(0) = 1$)
$T(n) = T(n/2) + c = T(n/4) + 2c = ... = T(1) + log_2(n)$
Therefore $T(n) = \Theta(log(n))$