3. T_A - function returning running time of algorithm A
T_B - function returning running time of algorithm B

$T_A(n) = 7\,T_A(\frac{n}{2}) + n^2$ $\qquad$ $T_B(n) = \alpha\,T_B(\frac{n}{4}) + n^2$

$k = \log_2 7 \quad 2^2 < 7 < 2^3$ $\qquad$ $k = \log_4 \alpha \quad 4^2 < \alpha < 4^3$

$= 2.80735$ $\qquad$ $16 \quad 49 \quad 64$

$f(n) = n^2$ $\qquad$ $f(n) = n^2$

$\rightarrow n^{\log_2 7}$ grows faster than $f(n)$

Thus $T_A(n) = \Theta(n^{\log_2 7})$

$\Rightarrow \log_2 7 = \log_4 49$

Thus if algorithm B is faster than A, α must be smaller than 49.

Therefore, the largest integer value of α is 48.