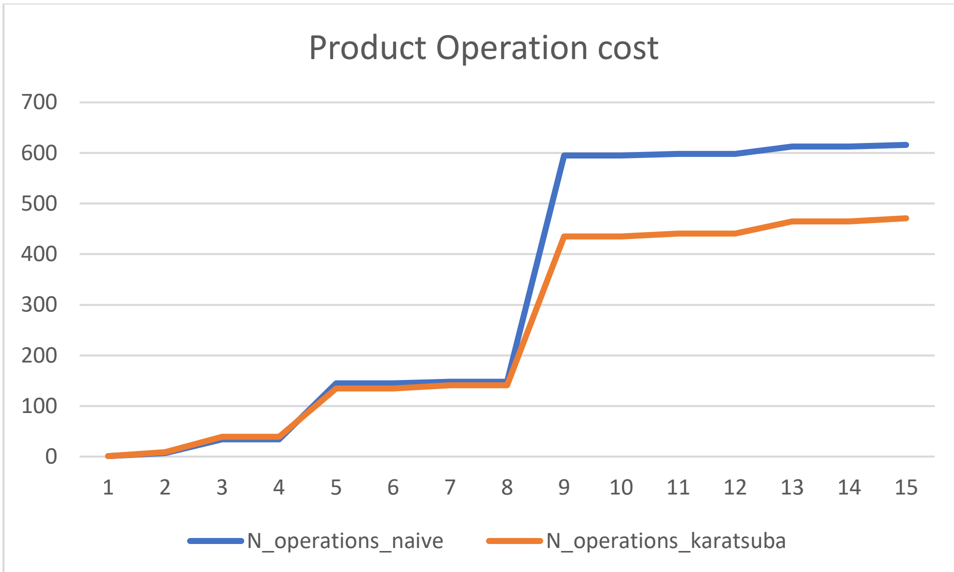
Comp 251: Assignment 4
Ashish Paul 260706034

1) The plot of the data in *karatsuba.csv* is shown below:



The number of operations for karatsuba algorithm is increasing faster than the naive algorithm for increasing larger integer bits. The graph flattens for both karatsuba and naive at around 9 bits because the number of recursive calls required does not increase when more bits are added.

2) **a.** $k = \log_5(25) = 2$

Case 1: $f(n) = n = O(n^{k-\varepsilon}) = O(n^{2-\varepsilon})$ is true for $0 < \varepsilon$ (e.g. $\varepsilon = 1$).

$\boldsymbol{T(n) = \Theta(n^2)}$

b. $k = \log_3(2)$

Case 3, part 1: $f(n) = n\log(n) = \Omega\left(n^{\log_3(2)+\varepsilon}\right)$ is true for $0 < \varepsilon$ (e.g. $\varepsilon = 0.1$).

Case 3, part 2: $2f\left(\frac{n}{3}\right) \le cf(n) \to 2\left(\frac{n}{3}\right)\log\left(\frac{n}{3}\right) \le cn\log(n) \to \frac{2}{3}\log(n) - \frac{2}{3}\log(3) \le$ $\log(n)$ is true for $c > \frac{2}{3}$ (for large *n*)

$\boldsymbol{T(n) = \Theta(n\log(n))}$

c. $k = \log_{\frac{3}{4}}(1) = 0$

Case 2: $f(n) = 1 = \Theta(n^0 \log^0 n) = \Theta(1)$ is true.

$\boldsymbol{T(n) = \Theta\left(n^0\, log^1(n)\right) = \Theta\left(log(n)\right)}$

d. $k = \log_3(7)$

Case 3, part 1: $f(n) = n^3 = \Omega\left(n^{\log_3(7)+\varepsilon}\right)$ is true for $0 < \varepsilon$ (e.g. $\varepsilon > 3 - \log_3(7)$).

Case 3, part 2: $7f\left(\frac{n}{3}\right) \leq cf(n) \rightarrow 7\left(\frac{n}{3}\right)^3 \leq cn^3 \rightarrow \frac{7}{27}n^3 \leq \text{cn}^3$ is true for $c < 1$

$\boldsymbol{T(n) = \Theta\left(n^3\right)}$

e. $k = \log_2(1) = 0$

Case 3, part 1: $f(n) = n(2 - \cos(n)) = \Omega(n^{0+\varepsilon})$ is true for $0 < \varepsilon$.

Case 3, part 2: Let $n = 2\pi m$, m odd. $1f\left(\frac{n}{2}\right) \leq cf(n) \rightarrow \frac{n}{2}\left(2 - \cos\left(\frac{n}{2}\right)\right) \leq c(n(2 - \cos(n)) \rightarrow$ $\frac{2\pi m}{2}\left(2 - \cos\left(\frac{2\pi m}{2}\right)\right) \leq c(2\pi m(2 - \cos(2\pi m)) \rightarrow 3\pi m \leq c(2\pi m)$ is only true for $c > \frac{3}{2} > 1$

The Master Theorem doesn't apply.

3) Apply the Master theorem to T_A:

$k = \log_2(7)$

Case 1: $f(n) = n^2 = O\left(n^{\log_2(7)-\varepsilon}\right)$ is true for $0 < \varepsilon < (2 - \log_2(7)$.

Therefore, $\boldsymbol{T_A(n) = \Theta(n^{log_2(7)})}$

Apply the start of the Master theorem to T_B. Case 1 also applies.

$k = \log_4(\alpha)$

Case 1: $f(n) = n^2 = O\left(n^{\log_4(\alpha)-\varepsilon}\right)$

Choose α to meet the following:

A: $\log_4(\alpha) > 2$ (to satisfy the Master theorem)

⇨ $\alpha > 4^2 = 16$

B: $\log_4(\alpha) < \log_2(7)$ (to be faster than T_A)

⇨ $\alpha < 4^{\log_2(7)} = 49$

Therefore, $16 < \alpha < 49$.

Choose the highest possible integer value of α, **which is 48**.

Therefore, the largest integer value of α for which T_B is asymptotically faster than T_A is **48**.