# Practice Assignment 4

Andrew Johnson

1) a) $T(n) = 2T(n/3) + 1 = \sum_{i}^{\log_3 n} c\left(\frac{n}{3^i}\right)^0 \cdot 2^i = c\sum 2^i = c\,\Theta(2^{\log_2 n}) = \Theta(n)$

b) $T(n) = 5T(n/4) + n = \sum^{\log_4 n} c\left(\frac{n}{4^i}\right) \cdot 5^i = cn\sum\left(\frac{5}{4}\right)^i = cn\,\Theta\left(\left(\frac{5}{4}\right)^{\log_2 n}\right)$

$c\,\Theta\left(\frac{5^{\log_2 n}}{n^k}\right) = \Theta\left(\frac{n^{\log_2 5}}{n}\right) = \Theta\left(n^{(\log_2 5 - 1)}\right)$

c) $T(n) = 7T(n/4) + n = \sum^{\log_4 n} c\left(\frac{n}{4^i}\right)4^i = \sum^{\log_2 n} cn = \Theta(n\log_2 n)$

d) $T(n) = 9T(n/3) + n^2 = \sum^{\log_3 n} c\left(\frac{n}{3^i}\right)^2 9^i = \sum^{\log_3 n} c\left(\frac{n^2}{3^{2i}}\right)9^i = \Theta(n^2 \log n)$

e) $T(n) = 8T(n/2) + n^2 = \sum^{\log_2 n} c\left(\frac{n}{2^i}\right)^3 8^i = \sum^{\log_2 n} c\left(\frac{n^3}{8^i}\right)8^i = \Theta(n^3 \log_2 n)$

f) $T(n) = T(n-1) + 2 \Rightarrow T(n-k) + 2\cdot k \;\; \forall k \rightarrow T(n-n) + 2n \quad T(0) = \Theta(1)$

$= T(0) + 2n = \Theta(n)$

g) $T(n) = T(n-1) + n^c = T(n-n) + n\cdot n^c = \Theta(n^{c+1}) \quad c \geq 1$

h) $T(n) = T(n-1) + c^n = T(n-n) + n\cdot c^n = \Theta(nc^n) \quad c > 1$

i) $T(n) = 2T(n-1) + 1 = 2(2^{n-1} - 1) + 1 = 2^n + 1 = \Theta(2^n)$

j) $T(n) = T(\sqrt{n}) + 1$ pick $n = 2^{2^m}$ so $\sqrt{n} = 2^{2^{m-1}}$

so going from $2^{2^0} \rightarrow 2^{2^m}$ it's incremented by $m$

so $T(n) = m$ and $m = \log_2(\log_2(2^{2^m}))$ and $n = 2^{2^m}$

so $T(n) = \Theta(\log_2(\log_2(n)))$

2) a) Start with an array
if the array has 2 numbers and the first is larger than the second
Swap the numbers
if it has more than 2 numbers
take $^2/_3$ of the size of the array
recursively call the first $^2/_3$
recursively call the last $^2/_3$
recursively call the first $^2/_3$ again
so it takes any large numbers from the beginning of the array and puts them in the middle, then it takes them and puts the small numbers from the last third and switches them with the larger numbers recently added or already in the middle, then it sees if the new small numbers from the last third are smaller than the newly made first third. If they are switch them.

b) If we used floor instead of ceiling, looking at n=4 the split will be 2 and 4
so the first and third recursive call won't look at the middle numbers of the array so it might not properly sort

c) $T(n) = 3T(\frac{3n}{2}) + \Theta(1)$

d) $T(n) = \sum_{i=0}^{\log_2 n} c\left(\frac{3n}{2}\right)^0 3^i = \Theta(3^{\log_2 n}) = \Theta\left(n^{\log_2 3}\right)$