Assignment 12

Alex Clemmer
Student number: u0458675

1

We can describe this specific case of merge sort with the recurrence relation $T(n) = 2T(n/2) + O(1)$. Since $2 > 2^0$, we know that $O(n^{log_2 2}) = O(n^1)$. Note that I used a generalization of the Master Theorem here.

2

We have n sequential elements for some arbitrarily large n. Since we are representing them in bits, the comparisons will take $O(n \log n)$ comparisons, because representing a number takes $\log_2$ bits, and you must compare each bit in the worst case for every n.

Unfortunately, this gives us an equation that doesn't really fit with the master theorem. What we do know is that $d > 1$. Since $\log_2 2 = 1$, and $d > 1$, we know that $O(n^d)$ for some $d > 1$.

3

What we want to do is find a quadratic (much like Gauss's example) whose terms cancel out and leave us with only the product we seek. This turns out to be extremely simple:

$$\begin{aligned} 0 &= (x+y)^2 - (x-y)^2 \\ 0 &= x^2 + 2xy + y^2 - x^2 + 2xy - y^2 \\ 0 &= 2xy + 2xy \end{aligned}$$

That turns out to be too much by a factor of 4. So our final equation will be the above divided by 4:

$$\frac{(x+y)^2 - (x-y)^2}{4} \tag{1}$$

See? Simple.