Clay Buxton **Homework 2** CS322 Algorithms
March 31, 2020

Problem 1

Algorithm 1: Divide and Conquer Finding Min and Max

```
DCMinMax(arr[], l, r, min, max){
        if(l == r){
                min = A[l], max = A[l]
        }
        else if(r-l = 1){
                if(A[l] == A[r]){
                        min = Al[l], max = A[r]
                }
                else{
                        min = A[r], max = A[l]
                }
        }
        else{
                DCMinMax(A[0,(l+r)/2],min,max) //Give half the array
                DCMinMax(A[(l+r)/2+1,r],min1,min2)
                if(min2 < min)
                        min = min2
                if(max2 > max)
                        max = max2
        }
```

Problem 2

Recurrence for Pan's algorithm: $M(n) = 143,640M(n/70)$

Using Master Theorem: $\theta(n^p)$ where $p = log_{70}143640 = 2.795$

Recurrence for Strassens algorithm $s = log_2 7 = 2.807$

$\therefore$ The recurrence for Pan's algorithm is slightly lower than Strassen's

Problem 3

See sorts.cpp

Problem 4

Prove $\sum_{h-1}^{i=0} 2(h-i)2^i = 2(n - log_2(n+1))$ where $n = 2^{k+1} - 1$
Substitute n $2(2^{h+1} - 1 - log_2(2^{h+1})) = 2(2^{h+1} - h - 2)$
$\sum_{i=0}^{h-1} 2(h-2)2^i$
$= \sum_{i=0}^{h-1} 2(h-2)2^i$
$= 2(\sum_{h-1}^{i=0} h2^i - \sum_{h-1}^{i=0} h2^i - \sum_{h-1}^{i=0} i2^i$
$= 2(h(2^h - 1) - (h-2)2^h - 2)$
$= 2(2^{h+1} - h - 2)$
$\therefore \sum_{h-1}^{i=0} 2(h-i)2^i = 2(n - log_2(n+1))$