Kelly Williams

Problem 1 Analysis

Linear Search

```
for (int i=0; i<n; i++) {
    if (val == a[i])
        return i; }
```

remember... $\left(\sum_{i=x}^{y} 1 = (y-x)+1\right)$

$$O_b + \sum_{i=0}^{n-1} (O_i + PO_s) \qquad \text{let } O_i + PO_s = O_{is}$$

$$= O_b + \sum_{i=0}^{n-1} O_{is} = O_b + O_{is}\underbrace{(n-1-0+1)}_{n} = O_{is}(n) + O_b = \underline{C_1 n + C_0}$$

behaves like an order N function: $f(n) = C_1 n + C_0$ $\quad O(n)$

Binary Search

```
int lowEnd = 0;
int highEnd = n-1;
do {
    int middle = (highEnd + lowEnd)/2;
    if (val == a[middle])
        return middle;
    else if (val > a[middle])
        lowEnd = middle+1;
    else
        highEnd = middle-1;
    } while (lowEnd <= highEnd);
```

array size = n

one loop: size $= \frac{n}{2}$

two loops: size $= \frac{n}{4}$

K loops: $1 = \frac{n}{2^K}$

$2^K = n$

$K = \log_2(n)$

$$O_b + \sum_{i=0}^{K} (O_i + PO_s) \qquad \text{let } O_i + PO_s = O_{is}$$

$$= O_b + \sum_{i=0}^{K} O_{is} = O_b + O_{is}\sum_{i=0}^{K} 1$$

$$= O_b + O_{is}(K-0+1) = O_b + O_{is}(K+1)$$

$$= \underbrace{O_b + O_{is}}_{C_0} + \underbrace{O_{is}}_{C_1} K = C_0 + C_1 K \qquad (K = \log_2(n))$$

$\Rightarrow f(n) = C_0 + C_1 \log_2(n)$

$O(\log_2 n)$