Question #1 Midterm

Linear Search

```
for (int i = 0; i < n; i++)
{
    if (arr[i] == x) { return i; }
    return -1;
}
```

$$O_b + \sum_{i=0}^{n-1} \left(O_i + PO_s \right)$$

$$O_i + PO_s = O_{is}$$

$$O_b + \sum_{i=0}^{n-1} \left(O_{is} \right)$$

$$O_b + (n-1-0)+1 \left(O_{is} \right)$$

$$O_b + (n) \left(O_{is} \right) \qquad c' = O_{is} \quad c = O_b$$

$$f_{is}(n) = c'n + c$$

$$f_{is}(n) = O(g(n)) \quad \lim_{n \to \infty} \left(\frac{f_{is}(n)}{g(n)} \right)$$

Let $g(n) = N_{\geq 0}$

$$\lim_{n \to \infty} \left(\frac{c'n}{n} + \frac{c}{n} \right) = c'$$

$$\Rightarrow 0 \leq \frac{f_{is}(n)}{g(n)} \leq c \quad \text{for} \quad N > N_0$$

Binary Search

```
int lowend = 0
int highend = n-1

do { int middle = (highEnd + lowEnd) / 2;
        if( val == a[middle] ) return middle;
    elseif (val > a[middle]) lowend = middle + 1;
    else highend = middle - 1
   }
   while(low end <= highEnd);

return -1;
```

PO_m = Probability of halving
PO_f = Prob. of finding element
O_b = Operations before

$$O_b \sum_{i=0}^{n-1} \left(\frac{1}{2} PO_m \right) PO_f$$

$$O_b + \frac{n}{2} PO_m + PO_f$$

As n gets smaller by factors of 2 the # of operations decreases exponentially so $O(\log(n))$