

Binary Search

index	value
0	3
1	6
2	10
3	15
4	18
5	20
6	22
7	25
8	30
9	37
10	38
11	50
12	55
13	60
14	64
15	100

binary search for 30
number of iterations

```
public int binarySearch(int[] A, int x) {  
    int lo = 0;  
    int hi = A.length - 1;  
    while (lo <= hi) {  
        int mid = lo + (hi - lo) / 2;  
        if (x < A[mid])  
            hi = mid - 1;  
        else if (x > A[mid])  
            lo = mid + 1;  
        else  
            return mid;  
    }  
    return(-1);  
}
```

Sheet1

Page 2

Sequential Search

index	value
0	38
1	6
2	60
3	50
4	37
5	20
6	64
7	25
8	30
9	18
10	3
11	15
12	55
13	10
14	22
15	100

sequential search for 30

best case

worst case

average case

```
public int sequentialSearch(int[] A, int x) {  
    for(int k=0; k<A.length; k++)  
        if (A[k]==x)  
            return(k);  
  
    return(-1);  
}
```

Sheet2