Rubin Peci
CS 102 - Data Structures
October 15th, 2019

A.1
a. Runtime of O(n^2) because it is a nested for loop.
b. Runtime of O(n) because it loops thru the list once.
c. Runtime of O(n^3) because there are 3 for loops.

A.2
a.
N = |1 1|
|0 1|

N^1 = |1 1|
|0 1|

N^2 = |2 1|
|1 1|

N^4 = |5 3|
|3 2|

N^8 = |34 21|
|21 13|

b.
Base Case:
Fib(0) is 0 and Fib(1) is 1
Assuming the following is true...
N^n = |F(n+1) F(n)|
|F(n) F(n-1)|
When N = |1 1|
|0 1|
N^2n = N^n * N^n = |F(n+1) F(n) | * |1 1| = |F(n+1)*1 + f(n)*0 F(n+1)*1 + f(n)| = the next term in the sequence
|F(n) F(n-1)| |0 1| |F(n-1)*1 + f(n-1)*0 F(n)*1 + f(n-1)*1|

A.3
a.
We didn't include removeTail, and it's not incldued in the java libraries because it is a problematic function that could lead to various
NullPointerExceptions and is a time consuming process for a singly linked list.
b.

```
int removeTail() {
        if (head == null) return Integer.MIN_VALUE;
        Node n = head.next;

        while(n.next.next != tail){
                n = n.next;
        }

        tail = n.next;
        tail.next = null;
        return tail.value;
}
```

c.

```
for(int i = 0; i < 3; i++) {
        tailList.removeTail();
        headList.removeTail();
}
tailList.show("tailList after 3 removeTail() is:");
```

```
headList.show("headList after 3 removeTail() is:");
```

A.4
Pat1 (solution):
"^(?!.*(r|0|H|\\-|wk)).*$"

Pat2 (false negative):
"e"

Pat3 (false positive):
".*zip"

Pat4
".*rar"