1 More, MORE, MOREEEE (Spring 2016, MT2)

For all the methods below, give the runtime in $\Theta(\cdot)$ notation as a function of N. Your answer should be simple, with no unnecessary leading constants or summations.

```java
public static void p1(int N) {
    for (int i = 0; i < N; i += 1) {
        for (int j = 1; j < N; j = j + 2) {
            System.out.println("hi !");
        }
    }
}
```

P1 answer: $\Theta(N^2)$

```java
public static void p2(int N) {
    for (int i = 0; i < N; i += 1) {
        for (int j = 1; j < N; j = j * 2) {
            System.out.println("hi !");
        }
    }
}
```

P2 answer: $\Theta(NlogN)$

```java
public static void p3(int N) {
    if (N <= 1) return;
    p3(N / 2);
    p3(N / 2);
}
```

P3 answer: $\Theta(N)$

```
1  public static void p4(int N) {
2      int m = (int)((15 + Math.round(3.2 / 2)) *
3          (Math.floor(10 / 5.5) / 2.5) * Math.pow(2, 5));
4      for (int i = 0; i < m; i++) {
5          System.out.println("hi");
6      }
7  }
```

P4 answer: $\Theta(1)$

```
1  public static void p5(int N) {
2      for (int i = 1; i <= N * N; i *= 2) {
3          for (int j = 0; j < i; j++) {
4              System.out.println("moo");
5          }
6      }
7  }
```

P5 answer: $\Theta(N^2)$

2   A Wild Hilfinger Appears! (Fall 2017, Final)

a. Given the following function definitions, what is the worst-case runtime for p(N)?
Assume h is a boolean function requiring constant time.

Answer: $\Theta(N^2)$

```
1  int p(int M) {
2      return r(0, M);
3  }
4
5  int r(int i, int M) {
6      if (i >= M) return 0;
7      if (s(i) > 0) return i;
8      return r(i + 1, M);
9  }
10
11 int s(int k) {
12     if (k <= 0) return 0;
13     if (h(k)) return k;
14     return s(k - 1);
15 }
```
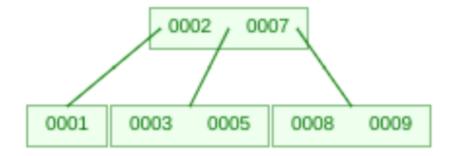
b. What is the worst-case runtime for the call p(N)? Assume that calls to h require constant time.

Answer: $\Theta(N^2)$

```
1   void p(int M) {
2       int L, U;
3       for (L = U = 0; U < M; L += 1, U += 2) {
4           for (int i = L; i < U; i+= 1) {
5               h(i);
6           }
7       }
8   }
```

3 Tree Time (Spring 2018, Midterm 2)

a. Draw the 2-3 tree that results from inserting 1, 2, 3, 7, 8, 9, 5 in that order.

b. Draw a valid BST of minimum height containing the keys 1, 2, 3, 7, 8, 9, 5.