1 Asymptotics is Fun!

(a) Using the function $g$ defined below, what is the runtime of the following function calls? Write each answer in terms of $N$.

```java
void g(int N, int x) {
    if (N == 0) {
        return;
    }
    for (int i = 1; i <= x; i++) {
        g(N - 1, i);
    }
}
```

$g(N, 1)$: $\Theta(\ )$

$g(N, 2)$: $\Theta(\ )$

(b) Suppose we change line 6 to $g(N - 1, x)$ and change the stopping condition in the for loop to $i \leq f(x)$ where $f$ returns a random number between 1 and $x$, inclusive. For the following function calls, find the tightest $\Omega$ and big $O$ bounds.

```java
void g(int N, int x) {
    if (N == 0) {
        return;
    }
    for (int i = 1; i <= f(x); i++) {
        g(N - 1, x);
    }
}
```

$g(N, 2)$: $\Omega(\ ), O(\ )$

$g(N, N)$: $\Omega(\ ), O(\ )$
Give the runtime of the following functions in $\Theta$ or $O$ notation as requested. Your answer should be as simple as possible with no unnecessary leading constants or lower order terms. For $f_5$, your bound should be as tight as possible (so don’t just put $O(N^{NM})$ or similar for the second answer).

```java
public static void f4(int N) {
    if (N == 0) {return;}
    f4(N / 2);
    f4(N / 2);
    f4(N / 2);
    f4(N / 2);
    g(N); // runs in $\Theta(N^2)$ time
}

Runtime: $\Theta(\ )$
```

```java
public static void f5(int N, int M) {
    if (N < 10) {return;}
    for (int i = 0; i <= N % 10; i++) {
        f5(N / 10, M / 10);
        System.out.println(M);
    }
}

Runtime: $O(\ )$
```
3 Flip Flop

Suppose we have the flip function as defined below. Assume the method unknown returns a random integer between 1 and N, exclusive, and runs in constant time. For each definition of the flop method below, give the best and worst case runtime of flip in \( \Theta() \) notation as a function of \( N \).

```java
public static void flip(int N) {
    if (N <= 100) {
        return;
    }
    int stop = unknown(N);
    for (int i = 1; i < N; i++) {
        if (i == stop) {
            flop(i, N);
            return;
        }
    }
}

(a) public static void flop(int i, int N) {
    flip(N - i);
}

Best Case: \( \Theta( ) \), Worst Case: \( \Theta( ) \)

(b) public static void flop(int i, int N) {
    int minimum = Math.min(i, N - i);
    flip(minimum);
    flip(minimum);
}

Best Case: \( \Theta( ) \), Worst Case: \( \Theta( ) \)

(c) public static void flop(int i, int N) {
    flip(i);
    flip(N - i);
}

Best Case: \( \Theta( ) \), Worst Case: \( \Theta( ) \)