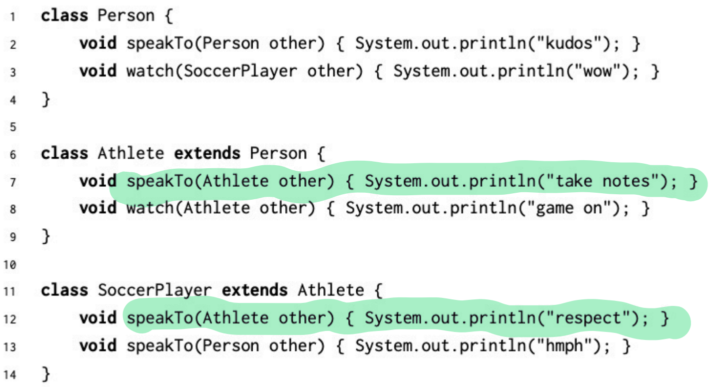
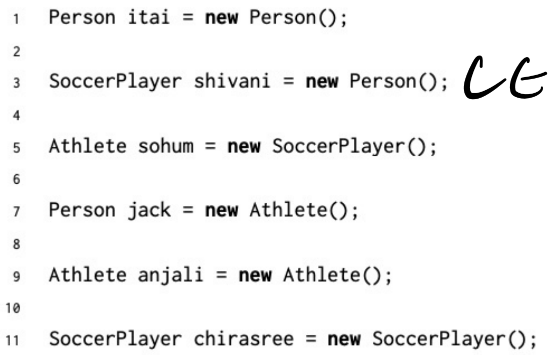


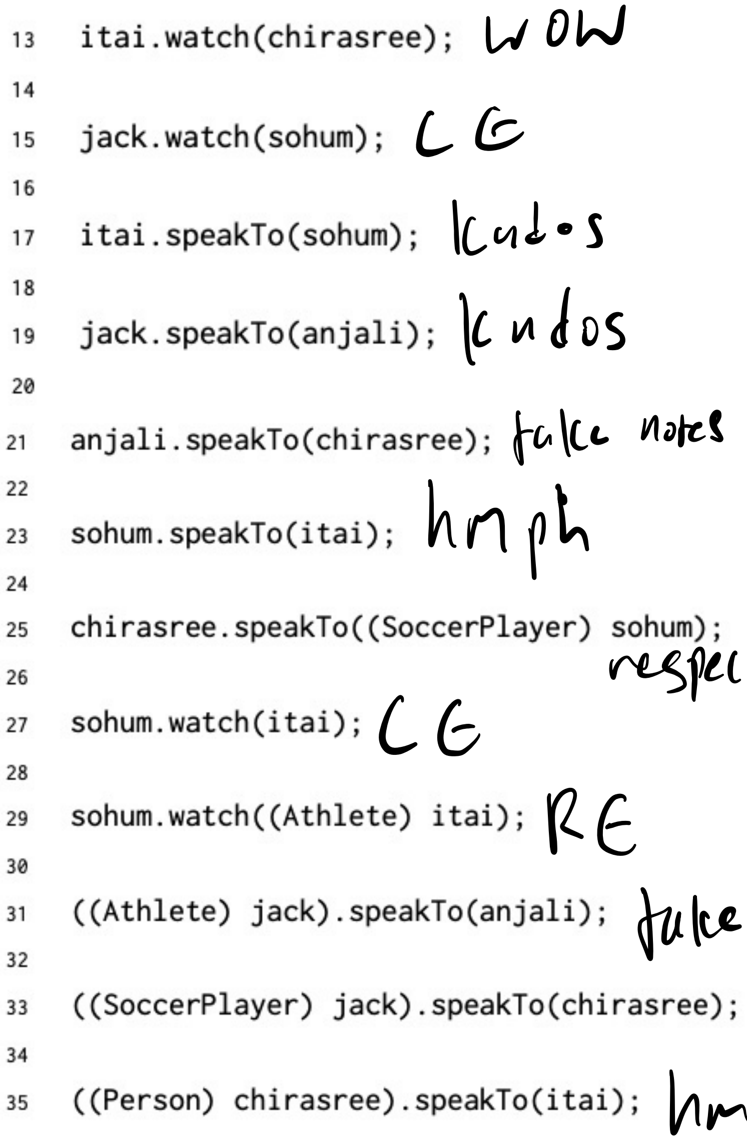
```java
class Person {
    void speakTo(Person other) { System.out.println("kudos"); }
    void watch(SoccerPlayer other) { System.out.println("wow"); }
}

class Athlete extends Person {
    void speakTo(Athlete other) { System.out.println("take notes"); }
    void watch(Athlete other) { System.out.println("game on"); }
}

class SoccerPlayer extends Athlete {
    void speakTo(Athlete other) { System.out.println("respect"); }
    void speakTo(Person other) { System.out.println("hmph"); }
}
```



```java
Person itai = new Person();

SoccerPlayer shivani = new Person();

Athlete sohum = new SoccerPlayer();

Person jack = new Athlete();

Athlete anjali = new Athlete();

SoccerPlayer chirasree = new SoccerPlayer();
```

CE

| Var | Static | Dynamic |
|---|---|---|
| itai | P | P |
| ~~shivani~~ | ~~SP~~ | ~~P~~ |
| sohum | A | SP |
| jack | P | A |
| anjali | A | A |
| Chirasee | SP | SP |



`itai.watch(chirasree);` wow

`jack.watch(sohum);` CE

`itai.speakTo(sohum);` kudos

`jack.speakTo(anjali);` kudos

`anjali.speakTo(chirasree);` take notes

`sohum.speakTo(itai);` hmph

`chirasree.speakTo((SoccerPlayer) sohum);` respect

`sohum.watch(itai);` CE

`sohum.watch((Athlete) itai);` RE

`((Athlete) jack).speakTo(anjali);` take notes

`((SoccerPlayer) jack).speakTo(chirasree);` RE

`((Person) chirasree).speakTo(itai);` hmph

Sp 18 MT2

Q8C

Suppose we have an algorithm with a runtime that is **$\Theta(N^2 \log N)$** in all cases.

Which of these statements are definitely **true** about the runtime, definitely **false**, or there is **not enough information (NEI)**?

| | | | |
|---|---|---|---|
| **$O(N^2 \log N)$** | [✓ ○ True] | ○ False | ○ NEI |
| **$\Omega(N^2 \log N)$** | [✓ ○ True] | ○ False | ○ NEI |
| **$O(N^3)$** | [✓ ○ True] | ○ False | ○ NEI |
| **$\Theta(N^2 \log4 N)$** | [✓ ○ True] | ○ False | ○ NEI |

↑

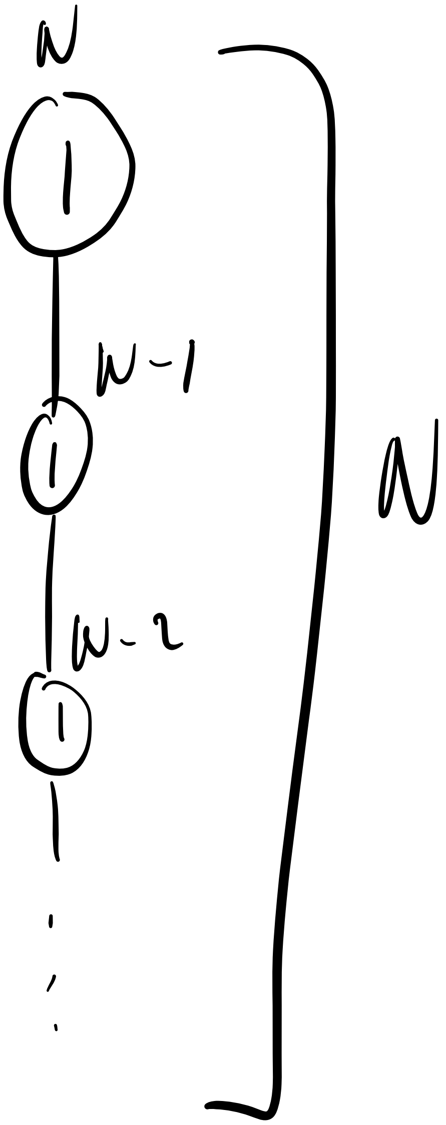
$\log_4 N$

all logs are in same family of functions

logs of diff bases differ by a constant,
so can treat all logs the same asymptotically

```
public static void g4(int N) {
    if (N == 0) {
        return;
    }

    g4(N - 1);

    if (k(N)) {
        g4(N - 1);
    }
}
```

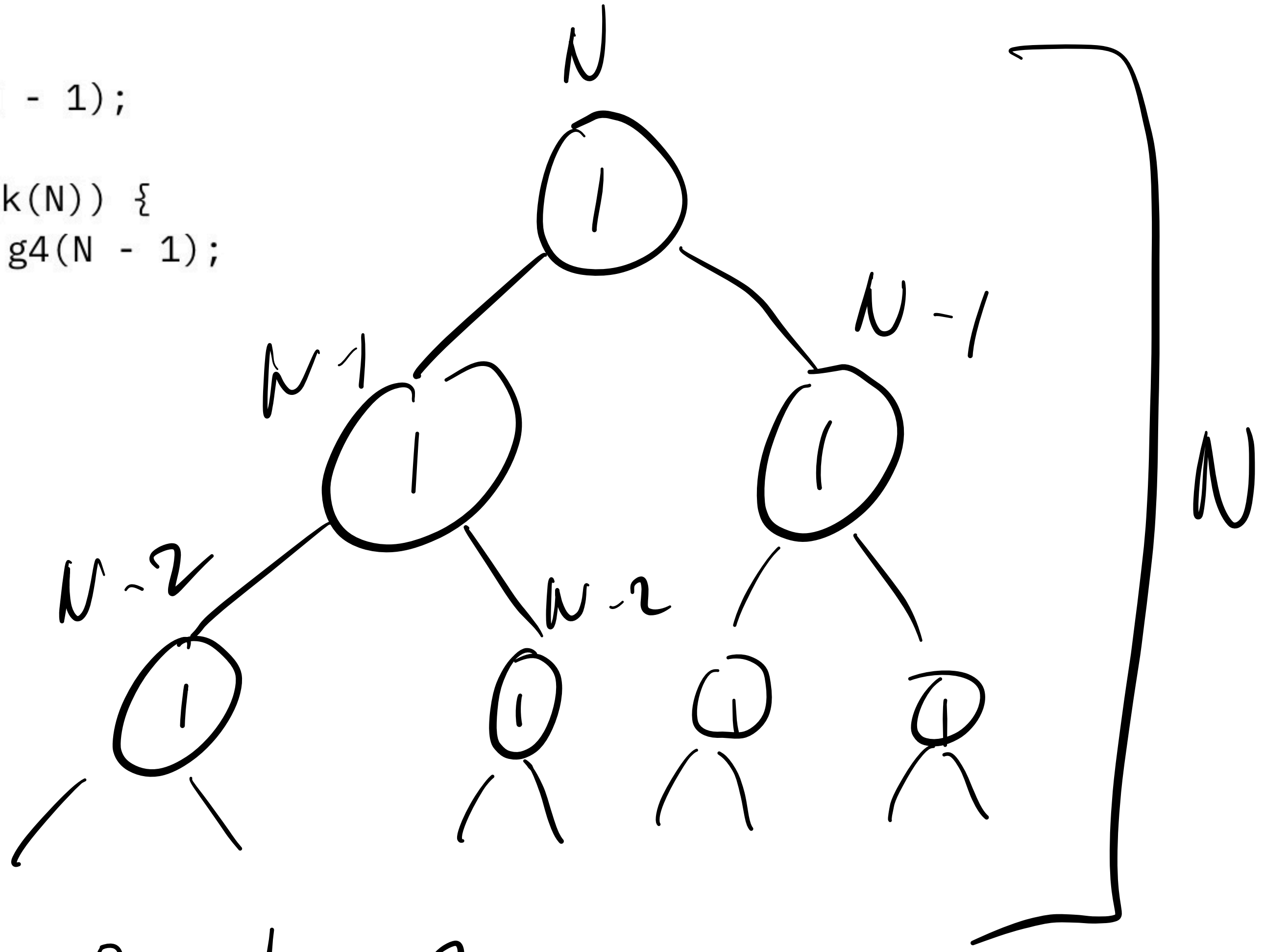
Best case: k(N) always false



$$\underbrace{1+1+1+\dots+1}_{N} = \Theta(N)$$

```
public static void g4(int N) {
    if (N == 0) {
        return;
    }

    g4(N - 1);

    if (k(N)) {
        g4(N - 1);
    }
}
```

Worst case: k(N) always true



$2^0 + 2^1 + 2^2 + ... + 2^N$

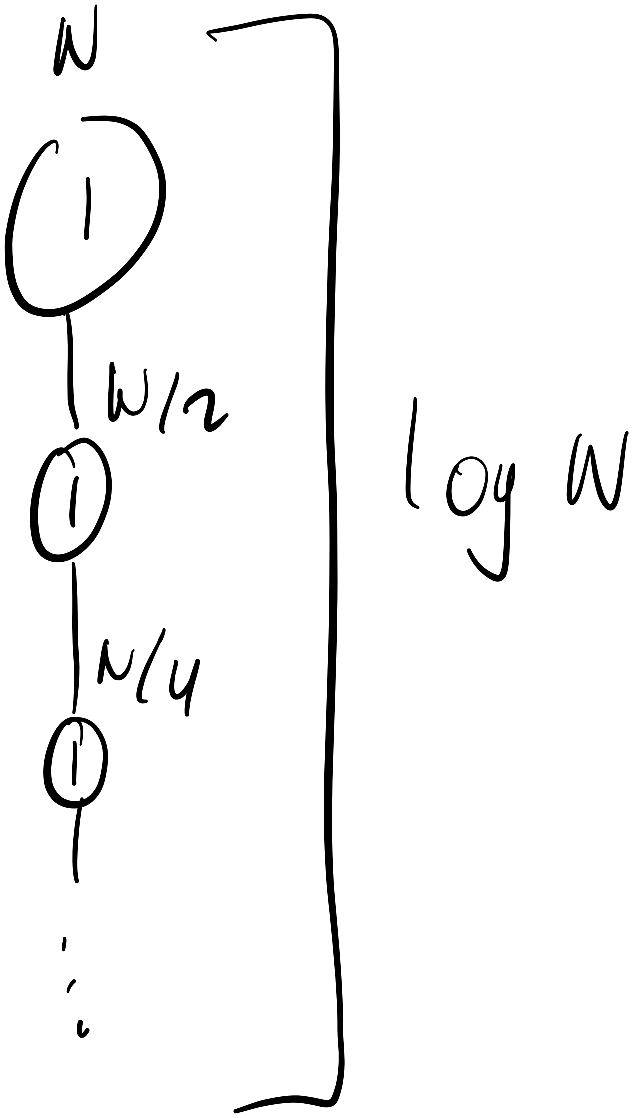
$1 + 2 + 4 + ... + 2^N = \Theta(2^N)$

$2^l$, $l$ = level

$1 + 2 + 4 + ... + f(N) = \Theta(f(N))$

```
public static void g5(int N) {
    if (N == 0) {
        return;
    }

    g5(N / 2);

    if (k(N)) {
        g5(N / 2);
    }
}
```

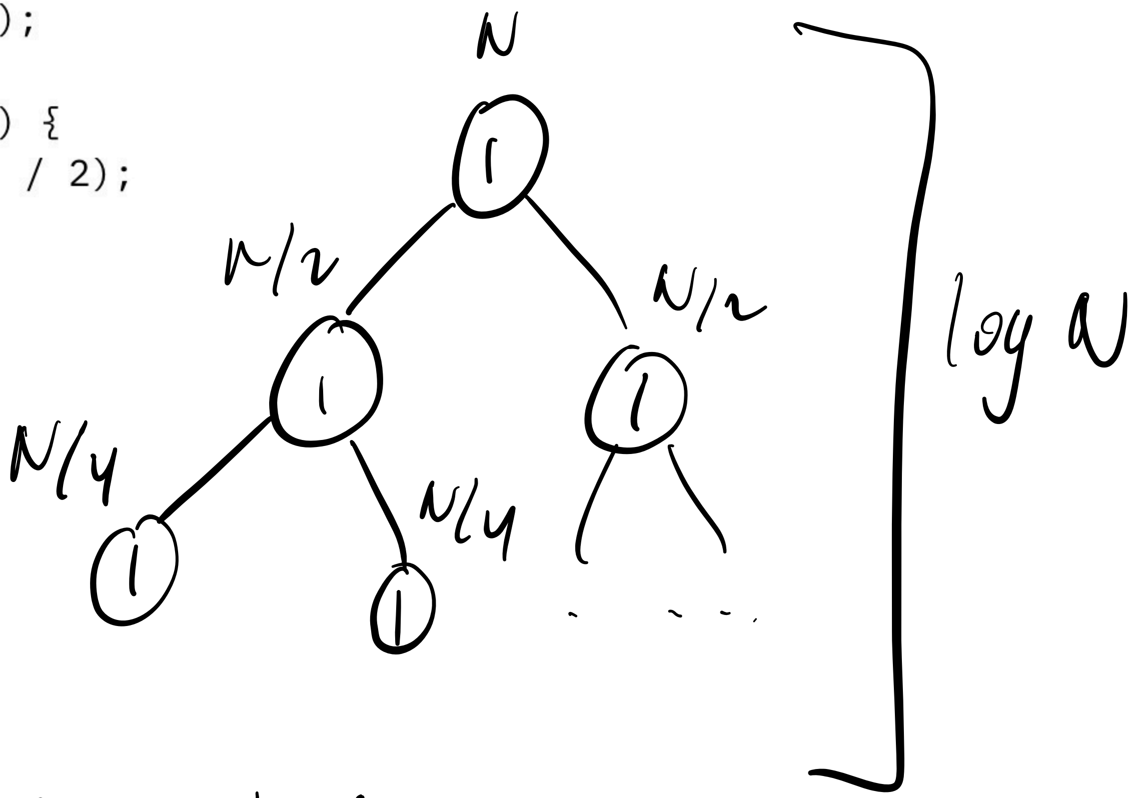
Besr



$$\underbrace{1+1+1+1\ldots+1}_{\log N\ \text{terms}} = \Theta(\log N)$$

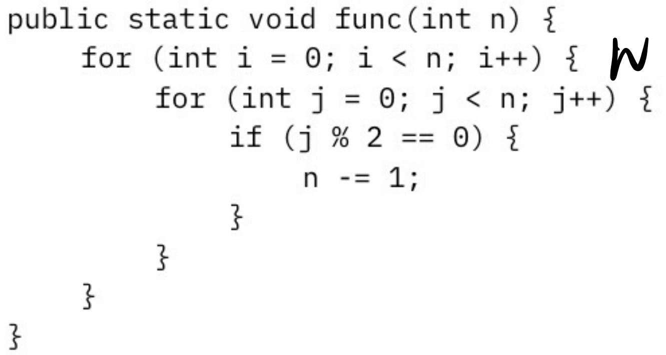
```
public static void g5(int N) {
    if (N == 0) {
        return;
    }

    g5(N / 2);

    if (k(N)) {
        g5(N / 2);
    }
}
```

Worst case



$$2^{\ell} = 2^{\log_2 N} = N$$

$$1 + 2 + 4 + \ldots + N = \Theta(N)$$



```
public static void func(int n) {
     for (int i = 0; i < n; i++) {
          for (int j = 0; j < n; j++) {
               if (j % 2 == 0) {
                    n -= 1;
               }
          }
     }
}
```

N

N

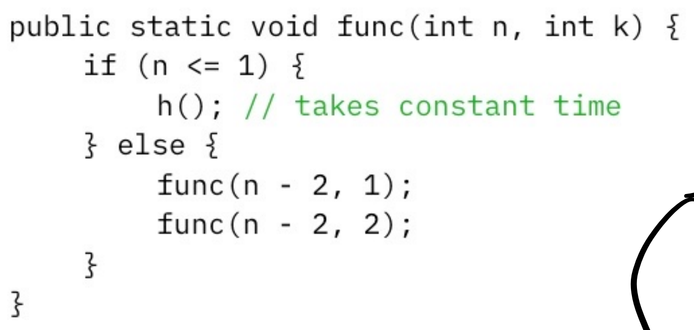
$$[0 \ldots N] = N/2 \text{ evens}$$

| inner loop runs | N |
|---|---|
| 1 | N/2 |
| 2 | N/4 |
| 3 | N/8 |
| ⋮ | N/N |

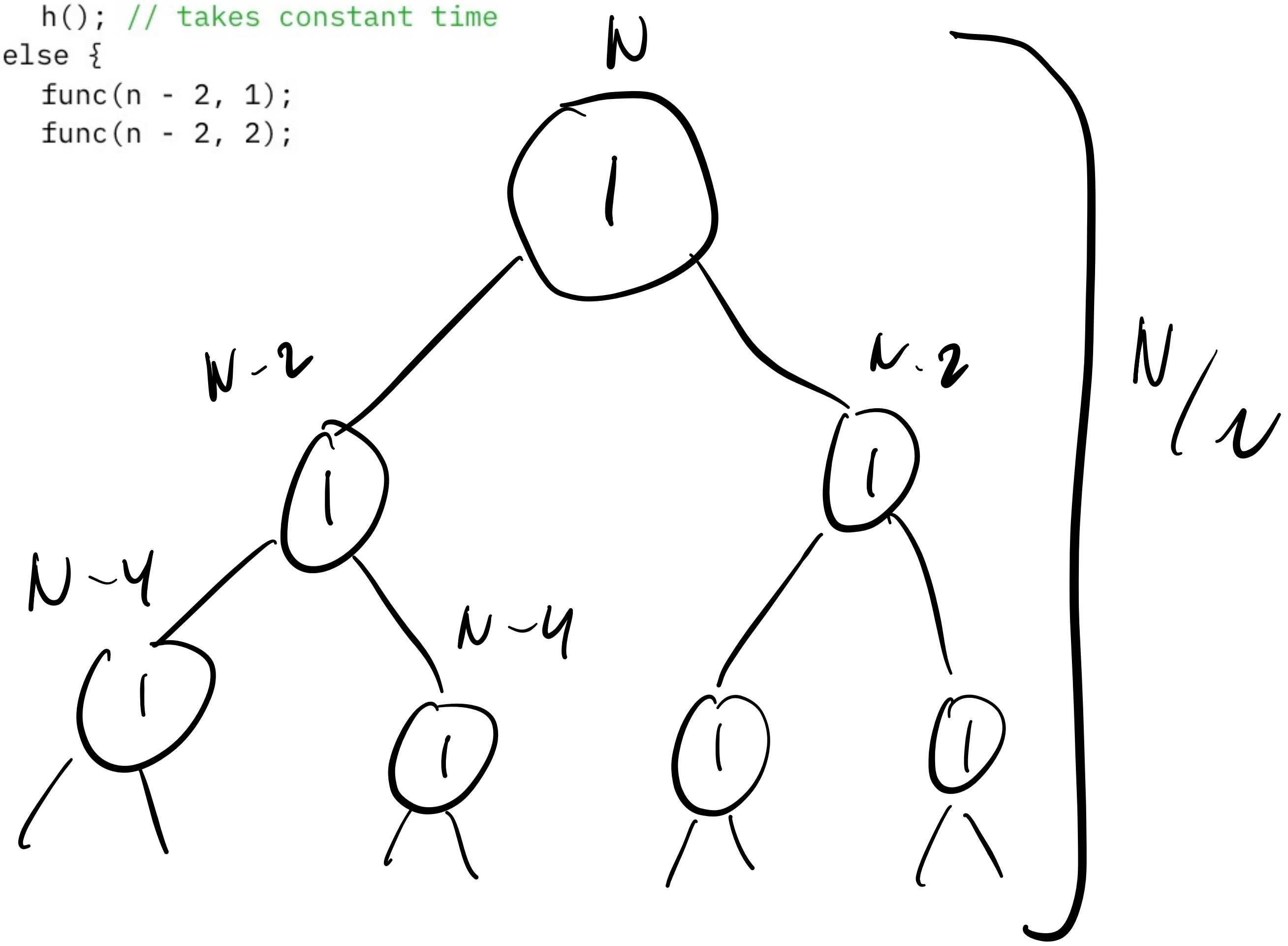
$$\frac{N}{2} + \frac{N}{4} + \frac{N}{8} + \ldots + \frac{N}{N}$$

$$N\left(\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \ldots + \frac{1}{N}\right)$$

$$N\left(\frac{1}{2}\right) = \Theta(N)$$



```java
public static void func(int n, int k) {
    if (n <= 1) {
        h(); // takes constant time
    } else {
        func(n - 2, 1);
        func(n - 2, 2);
    }
}
```



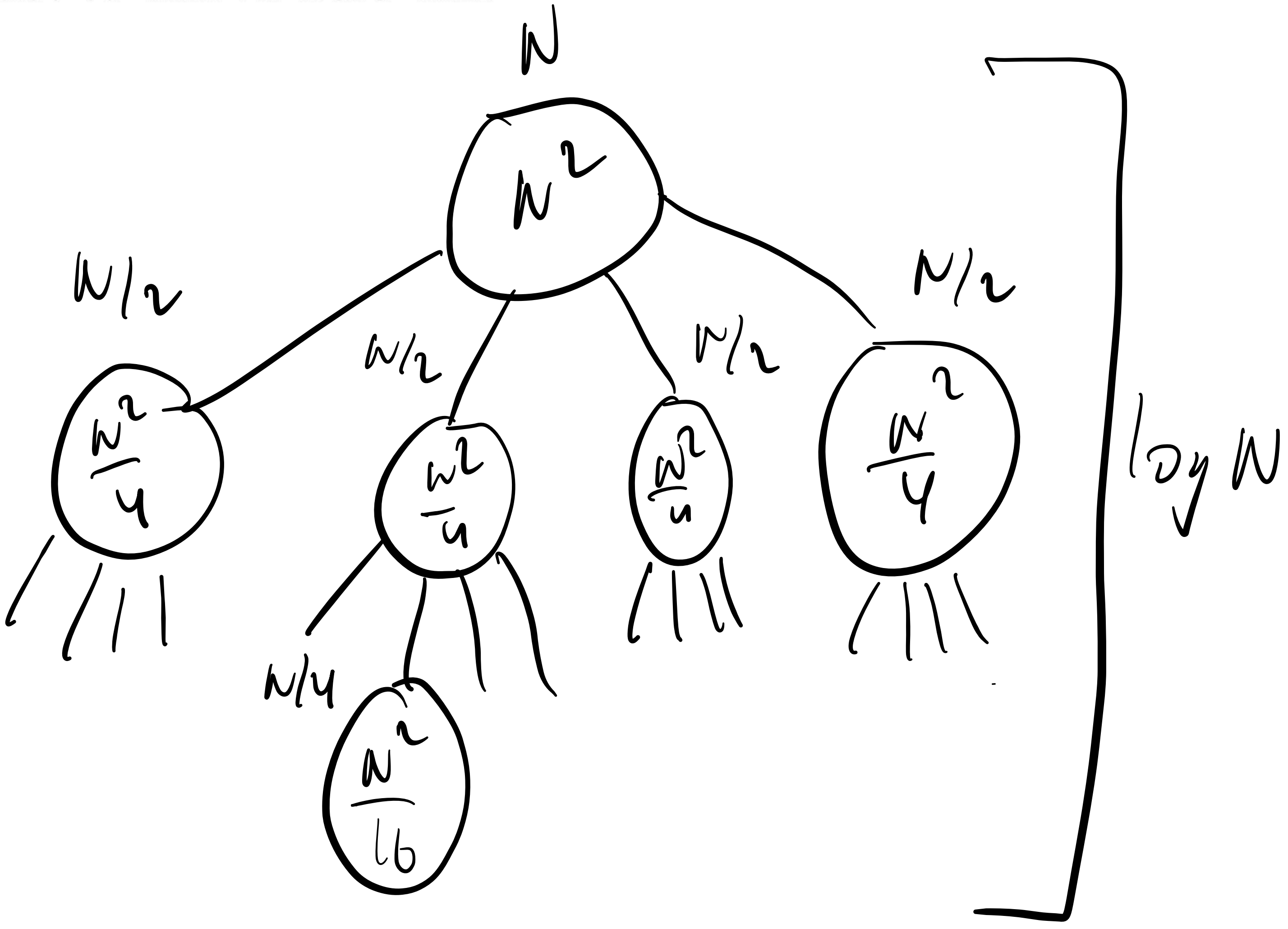
$$1+2+4+\dots+2^{N/2} = 2^{N/2}$$

$$= \Theta(2^N)$$

$$2^{\ell} = 2^{N/2}$$

$$\left(2^{N/2}\right)^2 = 2^n$$

```
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 Θ(N²) time
}
```



$$N^2 + 4\left(\frac{N^2}{4}\right) + 16\left(\frac{N^2}{16}\right) + \ldots +$$

$$N^2 + C_1\left(\frac{N^2}{C_1}\right) + C_2\left(\frac{N^2}{C_2}\right) + \ldots + C\left(\frac{N^2}{C}\right)$$

$$\underbrace{N^2 + C_1\left(\frac{N^2}{C_1}\right) + C_2\left(\frac{N^2}{C_2}\right) + \ldots + C\left(\frac{N^2}{C}\right)}_{\log N \text{ terms}}$$

$$\underbrace{N^2 + N^2 + N^2 + N^2 + \ldots + N^2}_{\log N \text{ terms}}$$

$$\Theta\left(N^2 \log N\right)$$