CS 61B Spring 2018 — Asymptotics

Exam Prep 7: February 26, 2018

1 It Begins (Spring 2017, MT2)

For each code block below, fill in the blank(s) so that the function has the desired runtime. Do not use any commas. If the answer is impossible, just write "impossible" in the blank.

```
public static void f1(int N) {       //Desired Runtime: Θ(N)
    for (int i = 1; i < N; i+=1) {System.out.println("hi");}
}
```

```
public static void f2(int N) {       //Desired Runtime: Θ(logN)
    for (int i = 1; i < N; i=i*2) {System.out.println("hi");}
}
```

```
public static void f3(int N) {       //Desired Runtime: Θ(1)
    for (int i = 1; i < N; i+=N) {System.out.println("hi");}
}
```

2 Slightly Harder (Spring 2017, MT2)

Give the runtime of the following functions in Θ or O notation as requested. Your answer should be as simple as possible with no unnecessary leading constatns or lower order terms. For f5, your bound should be as tight as possible (so don't just put $O(N^{NM!})$ or similar for the second answer).

```
Θ(N²logN) 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
          }
```

```
O(~~logN~~) N 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);
              }
          }
```

l b p

logN

$\underbrace{a\,b\;c\;d\;e}$

$e \cdot \overline{f(abcd)} + d \cdot \overline{f(abc)} + c\, f(ab) + a$

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

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

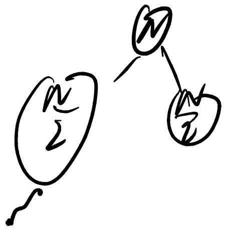
```java
public static void p1(int N) {
    for (int i = 0; i < N; i += 1) {
        for (int j = 1; j < N; j = j + 2) {
            System.out.println("hi !");
        }
    }
}
```

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: $\Theta(N \log N)$

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



$\log N$

$1+2+4+\cdots+2^H$

$2^{H+1}-1$

P3 answer: $\Theta(N)$

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

P4 answer: $\Theta(1)$

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

2^x

$1+2+4+\cdots+N^2$

$2^{x+1}-1$

2^N

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

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

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

Answer: $\Theta(N^2)$

```
int p(int M) {
    return r(0, M);
}

int r(int i, int M) {
    if (i >= M) return 0;
    if (s(i) > 0) return i;
    return r(i + 1, M);
}

int s(int k) {
    if (k <= 0) return 0;
    if (h(k)) return k;
    return s(k - 1);
}
```

$1+2+3+\dots+M$

b. What is the worse-case runtime for the call p(N)? Assume that calls to h require constant time.

Answer: $\Theta(N^2)$

```
void p(int M) {
    int L, U;
    for (L = U = 0; U < M; L += 1, U += 2) {
        for (int i = L; i < U; i+= 1) {
            h(i);
        }
    }
}
```

$\frac{M}{2}$ M-

2 4

$1+2+\dots+\frac{M}{2}$

+

1 2 2

2 4

3 6 3

$\frac{M-1}{\cancel{3}} + 1 \quad \frac{M-1}{2}$

$\frac{\quad}{2}$

$\frac{M-1}{2}$ $M-1$ $\frac{M-1}{2}$