CS 61B Spring 2025

Asymptotics II

Discussion 05: February 24, 2025

1 I am Speed

(a) 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. Assume that `System.out.println` runs in constant time. You may use Java's `Math.pow(x, y)` to raise `x` to the power of `y`.

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

```
// Desired Runtime: Theta(log N)
public static void f2(int N) {
    for (int i = 1; i < N; _i=log(i+1)___) {
        System.out.println("howdy Ergun");
    }
}
```

i*=2

```
// Desired Runtime: Theta(1)
public static void f3(int N) {
    for (int i = 1; _i<10_________; i += 1) {
        System.out.println("hello Anniyat");
    }
}
```

```
// Desired Runtime: Theta(2^N)
// This one is tricky! Hint: think about the dominating term in 1 + 2 + 4 + 8 + ...
+ f(N)
public static void f4(int N) {
    for (int i = 1; _i<2^N_____________; i *= 2) {
        for (int j = 0; j < i; j += 1) {
            System.out.println("what's up Alyssa");
        }
    }
}
```

Math.pow(2, N)

(b) *Extra:* Give the worst case and best case running time in $\Theta(\cdot)$ notation in terms of M and N. Assume that `kachow()` runs in $\Theta(N^2)$ time and returns a `boolean`.

```
for (int i = 0; i < N; i += 1) {
    for (int j = 1; j <= M; ) {
        if (kachow()) {
            j += 1;
        } else {
            j *= 2;
        }
    }
}
```

best:N3logM

worst:N3M

2 Re-cursed with Asymptotics!

(a) What is the runtime of the code below in terms of **n**?

```
public static int curse(int n) {
    if (n <= 0) {
        return 0;
    } else {
        return n + curse(n - 1);
    }
}
```

Theta(N)

(b) Can you find a runtime bound for the code below? We can assume the **System.arraycopy** method takes $\Theta(N)$ time, where N is the number of elements copied. The official signature is **System.arrayCopy(Object sourceArr, int srcPos, Object dest, int destPos, int length)**. Here, **srcPos** and **destPos** are the starting points in the source and destination arrays to start copying and pasting in, respectively, and **length** is the number of elements copied.

```
public static void silly(int[] arr) {
    if (arr.length <= 1) {
        System.out.println("You won!");
        return;
    }

    int newLen = arr.length / 2;
    int[] firstHalf = new int[newLen];
    int[] secondHalf = new int[newLen];

    System.arraycopy(arr, 0, firstHalf, 0, newLen);
    System.arraycopy(arr, newLen, secondHalf, 0, newLen);

    silly(firstHalf);
    silly(secondHalf);
}
```

Theta(NlogN)

(c) Given that **exponentialWork** runs in $\Theta(3^N)$ time with respect to input N, what is the runtime of **ronnie**?

```
public void ronnie(int N) {
    if (N <= 1) {
        return;
    }
    ronnie(N - 2);
    ronnie(N - 2);
    ronnie(N - 2);
    exponentialWork(N); // Runs in $Theta(3^N)$ time
}
```

可通过代码的递推式T(N)=3T(N−2)+3^N**展开**
然后算出总的时间（具体数学推导）

Theta(3^N)

3 Asymptotics Proofs

As a reminder, the formal definitions of Ω, Θ, and O are provided below:

Let f, g be real-valued functions. Then:

$f(x) \in \Theta(g(x))$ if there exists $a, b, N_0 > 0$ such that for all $N > N_0$, $|ag(N)| \leq |f(N)| \leq |bg(N)|$.

$f(x) \in O(g(x))$ if there exists $b, N_0 > 0$ such that for all $N > N_0$, $|f(N)| \leq |bg(N)|$.

$f(x) \in \Omega(g(x))$ if there exists $a, N_0 > 0$ such that for all $N > N_0$, $|ag(N)| \leq |f(N)|$.

Informally, we say that $f(x) \in O(g(x))$ approximately means that $f(x) \leq g(x)$, and similarly, $f(x) \in \Theta(g(x))$ means $f(x) = g(x)$ and $f(x) \in \Omega(g(x))$ means $f(x) \geq g(x)$. This problem will explore why we can make this informal statement, by showing that the O relation shares many properties with the $\leq$ relation.

For this problem, let f, g, and h be real-valued functions, and let x, y, and z be real numbers. You won't be expected to write full proofs on exams, but this thinking style will be helpful on exams and especially in later classes.

(a) If $x \leq y$, then $y \geq x$. Show that if $f(x) \in O(g(x))$, then $g(x) \in \Omega(f(x))$

original formula means there exists b, N0 such that |f(x)|<=|bg(x)|

both side divide b |1/bf(x)|<=|g(x)|

(b) If $x \leq y$ and $y \leq x$, then $x = y$. Show that if $f(x) \in O(g(x))$ and $g(x) \in O(f(x))$, then $f(x) \in \Theta(g(x))$

f(x)<=b1g(x) and g(x)<=b2f(x)
1/b2g(x)<=f(x)
so 1/b2g(x)<=f(x)<=b1g(x)

(c) For any real number, $x \leq x$. Show that for any function, $f(x) \in O(f(x))$.

f(x)<=1*f(x)

(d) If $x \leq y$ and $y \leq z$, then $x \leq z$. Show that if $f(x) \in O(g(x))$ and $g(x) \in O(h(x))$, then $f(x) \in O(h(x))$

(e) For any pair of real numbers x and y, either $x < y$, $x = y$, or $x > y$. Show that this is NOT a property of O; that is, find functions f and g such that $f(x) \notin O(g(x))$ and $g(x) \notin O(f(x))$.