CSCI3104
Spring 2023

Recitation: Asymptotic notation, time complexity of code

## Basics

Let $f, g : \mathbb{N} \to \mathbb{R}$ be sequences of real numbers. We say that $f(n) \in O(g(n))$ if there is a constant $c$ and a threshold natural number $N$ such that for all $n \geq N$, we have

$$f(n) \leq c \cdot g(n)$$

Let $f, g : \mathbb{N} \to \mathbb{R}$ be sequences of real numbers. We say that $f(n) \in \Omega(g(n))$ if there is a constant $c$ and a threshold natural number $N$ such that for all $n \geq N$, we have

$$f(n) \geq c \cdot g(n)$$

Let $f, g : \mathbb{N} \to \mathbb{R}$ be sequences of real numbers. We say that $f(n) \in \Theta(g(n))$ if it holds that both $f \in O(g)$ and $f \in \Omega(g)$ such that we can find $c_1$, $c_2$, $N$ which would satisfy for all $n \geq N\}$

$$0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n)$$

**Question 1.** Briefly justify that $\frac{1}{2}n^2 - 3n = \Theta\left(n^2\right)$.

$$\lim_{n\to\infty} \frac{\frac{1}{2}n^2 - 3n}{n^2} = \lim_{n\to\infty} \frac{\frac{\frac{1}{2}n^2 - 3n}{n^2}}{\frac{n^2}{n^2}} = \lim_{n\to\infty} \frac{\frac{1}{2} - \frac{3}{n}}{1} = \frac{\frac{1}{2} - 0}{1} = \frac{1}{2} \quad \#$$

**Question 2.** Next verify that $6n^3 \neq \Theta\left(n^2\right)$.

# Limit comparison test

If the limit $L := \lim_{x\to\infty} \frac{f(x)}{g(x)}$ exists, then:

1. If $0 < L < \infty$, then $f(n) \in \Theta(g(n))$.
2. If $L = 0$, then $f(n) \in O(g(n))$, but $f(n) \notin \Theta(g(n))$.
3. If $L = \infty$, then $g(n) \in O(f(n))$, but $g(n) \notin \Theta(f(n))$.

Also recall L'Hôpital's rule: suppose $g$ and $f$ are both differentiable functions, with either

- $\lim_{x\to\infty} f(x) = 0$ and $\lim_{x\to\infty} g(x) = 0$; or
- $\lim_{x\to\infty} f(x) = \pm\infty$ and $\lim_{x\to\infty} g(x) = \pm\infty$

If $\lim_{x\to\infty} \frac{f(x)}{g(x)}$ exists, then

$$\lim_{x\to\infty} \frac{f(x)}{g(x)} = \lim_{x\to\infty} \frac{f'(x)}{g'(x)}$$

**Question 3.** Order the two functions $f(n) = n^2 + 3n + 5$, and $g(n) = 3^n$.

$$\lim_{n\to\infty} \frac{n^2+3n+5}{3^n} \underset{LH}{=} \lim_{n\to\infty} \frac{2n+3}{3^n \ln 3}$$

$$\underset{LH}{=} \frac{1}{\ln 3} \lim_{n\to\infty} \frac{2}{3^n \ln 3} = 0.$$

**Question 4.** (1) Show that (a) $n^2 + 1 = \Omega(n)$, (b) $n^2 + 3n + 4 = \Theta\left(n^2\right)$.
(2) Order the two functions $n^2 \log(n)$ and $n(\log(n))^{10}$.
(3) Show that $\log_{\log 5}((\log n)^{100}) = \Theta(\log \log n)$.

(2). $$\lim_{n\to\infty} \frac{n^2 \lg(n)}{n(\lg(n))^{10}} = \lim_{n\to\infty} \frac{n}{(\lg_c n)^9} \underset{L'H.}{=} \lim_{n\to\infty} \frac{1}{9(\lg_c n)^8 \frac{1}{(\ln c)(n)}}$$

$$\underset{L'H}{=} \frac{\ln c}{9} \lim_{n\to\infty} \frac{n}{(\lg_c n)^8} \cdots \underset{L'H}{=} \cdots \underset{L'H}{=} \frac{(\ln c)^9}{9!} \lim_{n\to\infty} n = \infty$$

(3)

$$\lim_{n\to\infty} \frac{\lg_c(\lg_c n)}{\lg_c(\lg_c n)^{100}} = \lim_{n\to\infty} \frac{\lg_c \lg_c n}{100 \cdot \lg_c \lg_c n} = \frac{1}{100}$$ #

**Question 5.** Analyze the time complexity:

```
procedure Search(Array A[1, . . . , n], key)
    found ← false
    for i ← 1;i | len(A);i ← i + 1 do
        if A[i] == key then
            found ← true
return found
```

```
procedure foo_d(integer n):
   for i = 1, i <= n
       i = i * 2
       print 'outer'
       for j = 1, j <= i
           j = j + 2
           print 'inner'
```

Inner loop: Assume. # of iterations is $J$

$$J = \underset{d}{\arg\min} \; 1+2d$$

$$\text{s.t.} \quad 1+2d > n$$

$$\Rightarrow J \approx \frac{n-1}{2}$$

the time complexity for each iteration: 4

for inner loop: $\left(\frac{n-1}{2}\right) 4$

Outer loop: Assume # of iterations is $I$.

$$I = \underset{d}{\arg\min} \; 2^d$$

$$\text{s.t} \quad 2^d > n$$

$$\Rightarrow I \approx \lg_2 n$$

The time complexity of foo_d.

$$\sum_{i=1}^{\lg_2 n} \left(\frac{n-1}{2}\right) 4 + 4 = \sum_{i=1}^{\lg_2 n} \left(\frac{4}{2} n + \frac{4}{2}\right)$$

$$= 2(\lg_2 n)(n+1) = \Theta(n \lg n).$$

```
procedure foo_d(integer n):
    for i = 1, i <= n
        i = i*2
        print 'outer'
        for j = 1, j <= 2*i
            j = j+2
            print 'inner'
```

Inner loop: Assume. # of iterations is $J$

$$J = \underset{d}{\arg\min} \; 1+2d$$

$$\text{s.t.} \quad 1+2d > 2i$$

$$\Rightarrow J \approx \frac{2i-1}{2} = i - \frac{1}{2}$$

the time complexity for each iteration: 4

for inner loop: $4(i - \frac{1}{2})$.

Outer loop: Assume # of iterations is $I$.

$$I = \underset{d}{\arg\min} \; 2^d$$

$$\text{s.t.} \quad 2^d > n$$

$$\Rightarrow I \approx \lg_2 n$$

The time complexity of foo_d.

$$\sum_{i=1}^{\lg_2 n} 4(i - \tfrac{1}{2}) + 4 = \left(\sum_{i=1}^{\lg_2 n} 4i\right) + \frac{4}{2}(\lg_2 n)$$

$$= 4\left(\frac{(1+\lg_2 n)\lg_2 n}{2}\right) + \frac{4}{2}(\lg_2 n) = \Theta\left((\lg_2 n)^2\right)$$

# logarithm computation rules.

1. $\log_b(mn) = \log_b(m) + \log_b(n)$

2. $\log_b\left(\frac{m}{n}\right) = \log_b(m) - \log_b(n)$

3. $\log_b(n^P) = P \log_b n$

4. $\log_b(n) = \dfrac{\log_p(n)}{\log_p(b)}$

5. $\log_b(n) = \log_b(P) \cdot \log_p n$

6. $a^{\log_c b} = b^{\log_c a}$

we can prove rule 6:

Assume $a = X^n$, $b = X^m$

$$a^{\log b} = (X^n)^{\log X^m}$$

$$= (X^n)^{m \log X}$$

$$= X^{n \cdot m \log X}$$

$$= (X^m)^{n \log X}$$

$$= (X^m)^{\log X^n}$$

$$= b^{\log a}.$$

Differential rules.

1. $\frac{d}{dx} \log_a x = \frac{1}{x \ln(a)}$

2. $\frac{d}{dx} a^x = a^x \cdot \ln(a).$