50.004 Introduction to Algorithms, Fall 2018
Singapore University of Technology and Design
Gemma Roig, Ioannis Panageas, Subhajit Datta

18 Sep, 2018
Home Exercises - Week 1&2

Student Information

Name: Zhu Bo Student ID: 1002856
Due Date: 25 Sep, 11:59PM.
Submit answers on eDimension in pdf format. Submission without student information will **NOT** be marked! Any questions regarding the homework can be directed to the TA through email (contact information on eDimension).

Week 1

For all answers that are FALSE to a (T/F) question, please provide a short reason why as well.

1. The asymptotic complexity of $n^3 + 2n^2 + 1000$ is $O(n^3)$. **(T/F)** T highest power n^3
2. The asymptotic complexity of $100n^2 + n + \cos n + 1000$ is $\Theta(n^2)$. **(T/F)** T highest power n^2
3. The asymptotic complexity of $100n^{10} + n^{2.3} + 1000$ is $\Omega(n^9)$. **(T/F)** T highest power of n^{10} which is $\geq n^9$
4. The asymptotic complexity of $n^2 + n + 1000$ is $\Theta(n^{1.5})$. **(T/F)** F highest power n^2 which is $\geq n^{1.5}$ $\therefore \Theta(n^{1.5})$ cannot be true
5. Given a program that performs the following (assuming printing takes $\Theta(1)$):

   for(int $i = 0; i < n^2; i$*++)*
   for(int $j = 0; j < n; j$*++)*
   for(int $k = 0; k < 10; k$*++)*
   print(Hello)

   1st for loop $\Theta(n^2)$
   2nd for loop $\Theta(n)$
   3rd for loop $\Theta(1)$

   The asymptotic complexity is $\Theta(n^2)$. **(T/F)** F

   $\therefore \Theta(n^3)$ is the complexity.
6. Given a program that performs the following (assuming printing takes $\Theta(1)$):

   for(int $i = 0; i < 100; i$*++)*
   for(int $j = 0; j < n; j$*++)*
   print(Hello)

   1st for loop $\Theta(1)$
   2nd for loop $\Theta(n)$.

   The asymptotic complexity is $\Theta(n)$. **(T/F)** T

   $\therefore \Theta(n)$ is the complexity.
7. Given a program that performs the following (assuming printing takes $\Theta(1)$):

   for(int $i = 0; i < 100; i$*++)*

$$for(int\ j = 0;\ j < 500;\ j{+}{+})$$
$$\quad print(n)$$

both forloops are $\Theta(1)$
$\therefore$ complexity $= \Theta(1)$

The asymptotic complexity is $\Theta(n)$. **(T/F)** F

8. Given $f(n) = n^3 + n^2$ and $g(n) = 10n^2$, $f(n) = \Theta(g(n))$. **(T/F)** F $n^3 \geqslant n^2 \therefore$ not true.
9. Given $f(n) = n^{0.5} + 10$ and $g(n) = n + 10$, $f(n) = O(g(n))$. **(T/F)** T $n^{0.5} \leqslant n \therefore g(n)$ is always grow faster
10. The ranking of the functions below, sorted in **ascending** order of growth is (B).

    A. $n^2 < nlog(n) < 2^n < n^n$

    B. $nlog(n) < n^2 < 2^n < n^n$

    C. $nlog(n) < n^2 < n^n < 2^n$

    D. $n^2 < nlog(n) < n^n < 2^n$

according to the slides:
$n\log(n)$ grows slowest.
n^n grows fastest
$2^n > n^2$

Week 2

1) Use the Master Theorem to give tight asymptotic bounds for the following recurrences. Please show how you derive your answer.

1. $T(n) = 2T(n/4) + n^2$
2. $T(n) = 2T(4n/5) + \log n$
3. $T(n) = 2T(n/4) + \sqrt{n}$
4. $T(n) = \sqrt{2}T(n/4) + n \log n$

Q1: $T(n) = 2T(n/4) + n^2$

$\log_b a = \frac{1}{2}$

Apply master theorem case 3

$n^2 = \Omega(n^{\frac{1}{2}+\varepsilon})$

we choose $\varepsilon = \frac{3}{2}$ check:

$n^2 = \Omega(n^2)$ $\quad 2(\frac{n}{4})^2 \leq cn^2$

$T(n) = \Theta(f(n))$ $\quad \Leftrightarrow \frac{n^2}{8} \leq cn^2$

$= \Theta(n^2)$ $\quad$ choose $c = \frac{1}{8}$ $\forall n \geq 1$

it's true

$\therefore T(n) = \Theta(n^2)$

Q2: $T(n) = 2T(4n/5) + \log n$

$\log_b a = \log_{\frac{5}{4}} 2 = 3.106$

Apply master theorem case 1

$\log n = O(\log n^{3.106-\varepsilon})$

$T(n) = \Theta(n^{3.106})$

Q3: $T(n) = 2T(n/4) + \sqrt{n}$

$\log_b a = \log_4 2 = \frac{1}{2}$

Apply case 2

$f(n) = \Theta(n^{\log_b a})$

$n^{\frac{1}{2}} = \Theta(n^{\frac{1}{2}})$

Since this holds,

$T(n) = \Theta(n^{\log_b a} \log^{k+1} n)$

$k \geq 0$ we $k = 0$

$= \Theta(n^{\frac{1}{2}} \log n)$

Q4: $T(n) = \sqrt{2}\, T(n/4) + n \log n$

$\log_b a = \frac{1}{4}$

By case 3

$n \log n = \Omega(n^{\log_4 \sqrt{2} + \varepsilon})$

$n \log n = \Omega(n^{\frac{1}{4}+\varepsilon})$

let $\varepsilon = \frac{3}{4}$

$n \log n = \Omega(n)$.

check $\sqrt{2}(\frac{n}{2} \log \frac{n}{2}) \leq c n \log n$

when $c = \frac{\sqrt{2}}{2}$

it is always true that $\frac{\sqrt{2}}{2} n \log \frac{n}{2} \leq \frac{\sqrt{2}}{2} n \log n$

$\therefore T(n) = \Theta(n \log n)$