CSCE 222 [Sections 503, 504] Discrete Structures for Computing
Fall 2019 – Hyunyoung Lee

Problem Set 4

> **Due dates:** Electronic submission of *yourLastName-yourFirstName-hw4.tex* and *yourLastName-yourFirstName-hw4.pdf* files of this homework is due on **Monday, 9/30/2019 before 10:00 p.m.** on `http://ecampus.tamu.edu`. You will see two separate links to turn in the .tex file and the .pdf file separately. Please do not archive or compress the files. **If any of the two submissions are missing, you will likely receive zero points for this homework.**

Name: Ian Stephenson **UIN: 927004123**

Resources. (Peer Teacher Central, Discrete Math and Its Applications)

On my honor, as an Aggie, I have neither given nor received any unauthorized aid on any portion of the academic work included in this assignment. Furthermore, I have disclosed all resources (people, books, web sites, etc.) that have been used to prepare this homework.

Electronic Signature: **Ian Stephenson**

Total 100 points.

Problem 1. (5 points) Let x be a real number and n an integer. Show that

$$\lceil x \rceil = n \text{ if and only if } x \leqslant n < x + 1.$$

[Hint: Use the definition for $\lceil x \rceil = n$ given as the second fact among the four facts in slide #39 in the lecture slides on Sets and Functions. What you are proving here is the last fact in the same slide.]

Solution.

$n - 1 \leqslant x \leqslant n$ by the definition of ceiling
$\equiv n - 1 < x \wedge x \leqslant n$ by seperating the inequality
$\equiv n < x + 1 \wedge x \leqslant n$ by adding one to n - 1 and x
$\equiv x \leqslant n < x + 1$ by combining the two inequalities back into one statement

Problem 2. (15 points) Let n be a positive integer. Show that if n is a perfect square, then

$$\lfloor \sqrt{n} \rfloor - \lfloor \sqrt{n-1} \rfloor = 1.$$

[Hint: Use the definition for $\lfloor x \rfloor = n$ given as the first fact among the four facts in slide #39 in the lecture slides on Sets and Functions.]

Solution. Because n is a perfect sqaure, we can say that n = n^2. From the equation given, if we substitute in m^2 for n, we will get $\lfloor \sqrt{m^2} \rfloor - \lfloor \sqrt{m^2-1} \rfloor = 1$. $\lfloor \sqrt{m^2} \rfloor$ will become m because the square root of m^2 is an is m, and because m is an integer, the floor of an integer is just that integer. So our equation can be simplified down to be $m - \lfloor \sqrt{m^2-1} \rfloor = 1$. If we move the terms around, we get that $\lfloor \sqrt{m^2-1} \rfloor = m-1$. From the definition of floor, we could write this equation as an inequality, that is $m - 1 \leqslant \sqrt{m^2-1} < m$. This inequality can be proven by proving each part of the inequality. First, we must show that $m - 1 \leqslant \sqrt{m^2-1}$. If we square both sides and expand them out, we will get that $m^2 - 2m + 1 \leqslant m^2 - 1$. Solving this inequality for m, we get that $m \geqslant 1$. Because $m = sqrtn$ and n is always positive, then m will always have a solution of a positive integer, thus this statement is always true. The other part of this inequality is $m > sqrtm^2 - 1$. Squaring both sides, the inequality becomes $m^2 > m^2 - 1$ which simplifies to be $1 > 0$, which is always true. Therefore, because both parts of the inequality which represent the original theorem are true, the original theorem must also be true.

Problem 3. (10 points) Let f_1, f_2, f_3 be functions from the set $\mathbf{N}$ of natural numbers to the set $\mathbf{R}$ of real numbers. Suppose that $f_1 = O(f_2)$ and $f_2 = O(f_3)$. Is it possible that

$$f_1(n) > f_3(n)$$

holds for all natural numbers n? Give an example for f_1, f_2, and f_3 if it is possible, or give an explanation (convincing argument or proof) if it is impossible. [Hint: Think carefully about the *definition* of a function being in big-Oh of another function *as given in class*.]

Solution. It is impossible for $\mathrm{f_1(n)} > \mathrm{f_3(n)}$. By the definition of Big Oh and given the definitions listed above for f_1, f_2, and f_3, it would follow that $|f_1(n)| \leqslant U_1|f_2(n)|$ and $f_2(n) \leqslant U_2|f_3(n)|$. Because $f_1 < f_2 < f_3$, it would follow that $f_1 < f_3$. This makes sense because if f_2 upper bounds f_1, then f_3, which upper bounds f_2, should also upper bound f_1. We could then apply the definiton of Big Oh to f_1 and f_3. We could say that $|f_1| \leqslant U|f_3|$. From this, and from f_1 being upper bounded logically by f_3, it would follow that $f_3 > f_1$, and thus $f_1 > f_3$ is false.

Problem 4. (10 pts $\times$ 3 = 30 points) Determine whether each of the following statements is true or false. In each case, answer true or false, and justify your answer (by giving a direct proof if it is true, or a proof by contradiction if it is false; always use the definition involving the absolute values, as given in class).

a) $3n^2 + 41 = O(n^3)$

Solution. TRUE: By the definition of $f(n) = O(g(n))$, $|f(n)| \leqslant U|g(n)|$, where U is some positve real constant and $n \geqslant n_0$. From the question, we can say that $f(n) = 3n^2 + 41$ and $g(n) = n^3$. We can then say that $|3n^2 + 41| \leqslant U|n^3|$. To prove that this statement is true, we must show that at least one U with a cooresponding n_0, where all values of $n \geqslant n_0$, will make this expression evaluate to true. Lets say that $n_0 = 1$ and substitute this into the inequality. We will get that $|3(1)^2 + 41| \leqslant U|(1)^3|$, or that $|44| \leqslant U|1|$. We can drop the absolute values because all the terms are positive, thus getting that $44 \leqslant U$. From this inequality, we can say that it will be true for at least one n_0 and at least one U, that is that $U = 44$ and $n_0 = 1$, thus proving that $3n^2 + 41 = O(n^3)$.

b) $n^3 + 2n + 3 = O(n^2)$

Solution. FALSE: For $f(n) = O(g(n))$ to be true, the limit as n tends to infinity of $\frac{|f(n)|}{|g(n)|}$ must exist. If we set up the limit, we will get that $\lim_{n\to\infty} \frac{|n^3+2n+3|}{|n^2|}$. This limit will evaluate to infinity due to the power of higher order being in the numerator of the limit. However, there are two conditions to evaluate when determining whether or not a function is in Big Oh of another. The limit must exist, which this does, and the limit must be equal to some real number L, such that the upper bounding constant will be equal $L + / - \varepsilon$. This limit will fail the second condition because infinity is not a real number. Therefore, $f(n) \neq O(g(n))$.

c) $\frac{1}{2}n^2 + 5 = \Omega(n)$

Solution. TRUE: By the definition of $f(n) = \Omega(g(n))$, $L|g(n)| \leqslant |f(n)|$, where L is some positve real constant and $n \geqslant n_0$. From the question, if we take $f(n) = \frac{1}{2}n^2 + 5$ and $g(n) = n$, then $L|n| \leqslant |\frac{1}{2}n^2 + 5|$. If we take $n_0 \leqslant 1$, then the equation becomes $L|1| \leqslant |\frac{1}{2}1^2 + 5|$, or that $L \leqslant \frac{11}{2}$. By the definition of $f(n) = \Omega(g(n))$, there exists some constant L and some n_0, where for all n the inequality will hold, in this case $\frac{11}{2}$ and 1, such that the inequality $L|g(n)| \leqslant |f(n)|$ is true. Therefore, we have proven that $\frac{1}{2}n^2 + 5 = \Omega(n)$ is true.

Problem 5. (10 points) Let k be a fixed positive integer. Show that

$$1^k + 2^k + \cdots + n^k = O(n^{k+1})$$

holds.

Solution. From the definition of Big Oh, we could say that $|1^k+2^k+\cdots+n^k| \leqslant U|n^{k+1}|$. If we factor out an n from n^{k+1} and say that U = n, we will get that $|1^k+2^k+\cdots+n^k| \leqslant n|n^k|$. From this we could say that our $g(n) = n^k$. We could expand the sequence on the left to be $|1^k + 2^k + \cdots + (n-1)^k + n^k|$. We could then set $f(n) = (n-1)^k$. Thus, the statement $f(n) < g(n)$ will be true for all values of $n \geqslant 1$. We could now apply the theorem that if f(n) < g(n), then f(n) + g(n) = O(g(n)). This shows that for f(n) + g(n) $\leqslant$ U(g(n)). If we say that U = n and plug back in for f(n) and g(n), then we get that $(n-1)^k + n^k \leqslant n^{k+1}$. Because we added the two largest terms in the sequence of the original theorem, it would follow that for the same logic works for all values from 1^k to $(n-1)^k$. This proves that $1^k + 2^k + \cdots + n^k = O(n^{k+1})$ holds.

Problem 6. (15 points) Let f_1, f_2, f_3, f_4 be functions from the set $\mathbf{N}$ of natural numbers to the set $\mathbf{R}$ of real numbers. Suppose that $f_1 = O(f_2)$ and $f_3 = O(f_4)$. Use the *definition* of Big Oh *given in class* to prove that

$$f_1(n) + f_3(n) = O(\max(|f_2(n)|, |f_4(n)|)).$$

Solution. From the definition of Big Oh, $|f_1| \leqslant U_1|f_2|$, such that U_1 is a positive real constant and there exists some $n_{01} \leqslant n_1$ that makes the inequality true, and $|f_3| \leqslant U_2|f_4|$, such that U_2 is also a constant and there exists some $n_{02} \leqslant n_2$. We can say that the maximum value of n_{01}, n_{02} is equal to some n_{0x}. We can also let the maximum of the constants U_1, U_2 be equal to some U_x. Having these max values, we can establish an inequality from our definitions, $f_1(n) + f_3(n) \leqslant \max(|f_2(n)|, |f_4(n)|) \leqslant U_x(\max(|f_2(n), |f_4(n)|)$, which can be simplified to be $f_1(n) + f_3(n) \leqslant U_x(\max(|f_2(n), |f_4(n)|)$. This inequality fits the definition of Big Oh, as there is some positive constant, U_x and some n_0, n_{0x} such that the inequality holds true for all $n \geqslant n_{0x}$. Therefore, $f_1(n) + f_3(n) = O(\max(|f_2(n)|, |f_4(n)|))$. is true.

Problem 7. (5 pts $\times$ 3 = 15 points) Suppose that you have two algorithms A and B that solve the same problem. Algorithm A has worst case running time $T_A(n) = 2n^2 - 2n + 1$ and Algorithm B has worst case running time $T_B(n) = n^2 + n - 1$.

a) Show that both $T_A(n)$ and $T_B(n)$ are in $O(n^2)$.

b) Show that $T_A(n) = 2n^2 + O(n)$ and $T_B(n) = n^2 + O(n)$.

c) Explain which algorithm is preferable.

Solution. a) From the given equation $T_A(n)$, we can say that $|2n^2 - 2n + 1| \leqslant U|n^2|$. For $n_0 = 1$, we could with certainty that $2n^2 - 2n + 1 \leqslant 2n^2 + 0n^2 + 1n^2$ because each coeffecient on the right is either the same of is greater than the left, and the degree of n is greater. This inequality could be simplified to $2n^2 - 2n + 1 \leqslant 3n^2$. From the definition of Big Oh, we could say that $U = 3$ and $n \geqslant 1$, and therefore $T_A(n) = O(n^2)$. The same process goes for $T_B(n)$, but the inequality will be $|n^2 + n - 1| \leqslant U|n^2|$. Again, we can say that the inequality $n^2 + n - 1 \leqslant n^2 + n^2 + 0n^2$ will always be true. Simplify the expression and get that $n^2 + n - 1 \leqslant 2n^2$. We can now say that $U = 2$ and $n \geqslant 1$, and therefore from the definition of Big Oh, we can say that $T_B(n) = O(n^2)$.

b) To show that $T_A(n) = 2n^2 + O(n)$, we must show that $-2n + 1 = O(n)$. If we take the limit as $\lim_{n\to\infty} \frac{|-2n+1|}{|n|}$. This limit will evaluate to 2, and therefore $-2n+1 = O(n)$ is true. We can take the same process for $T_B(n) = n^2 + O(n)$ with $n - 1 = O(n)$. The limit $\lim_{n\to\infty} \frac{|n-1|}{|n|}$ will evaluate to 1, and therefore $n - 1 = O(n)$ is true.

c) The algorithm $T_B(n)$ is preferable because when n gets really large, the term that matters most is the term with the greatest power, and the coefficient on the largest term, n^2, for $T_B(n)$ is smaller than the coefficient for $T_A(n)$. Essentially, when n gets to be really big, $T_A(n)$ will run for approximately twice as long as $T_B(n)$.

Checklist:

- ☐ Did you type in your name and UIN?
- ☐ Did you disclose all resources that you have used?
  (This includes all people, books, websites, etc. that you have consulted.)
- ☐ Did you electronically sign that you followed the Aggie Honor Code?
- ☐ Did you solve all problems?
- ☐ Did you submit both of the .tex and .pdf files of your homework to the correct link on eCampus?