Problem set 2, problem 1

General comments

This problem was intended to give you practice writing proofs using the definitions of O and Ω.

Some of you made arguments using a kind of informal "big-O arithmetic". For example, you can make an argument that the statement $O(n) + O(n) = O(n)$ is both meaningful and true. But putting such a statement to use requires that you have both a clear definition of what the plus-sign means in $O(n) + O(n)$ and a clear definition of what it means to assert that two O-expressions are equal. Then, finally, you need a proof of your assertion based on the definition of O. Chapter 2 of Kleinberg & Tardos does not try to formalize big-O arithmetic in this way.

So let's go back to the definitions of O, Ω, and O for 2-variable functions.

Definition 0.1. *Given two functions* $f, g : \mathbb{N} \to \mathbb{R}^+$*, we say that* $\mathbf{f} = \mathbf{O}(\mathbf{g})$ *if* $\exists C > 0, N > 0$ *such that* $n \geq N \implies f(n) \leq Cg(n)$.

Definition 0.2. *Given two functions* $f, g : \mathbb{N} \to \mathbb{R}^+$*, we say that* $\mathbf{f} = \mathbf{\Omega}(\mathbf{g})$ *if* $\exists\ C > 0, N > 0$ *such that* $n \geq N \implies f(n) \geq Cg(n)$.

Definition 0.3. *Let* $f, g : \mathbb{N} \times \mathbb{N} \to \mathbb{R}^+$*. Then* $\mathbf{f}(\mathbf{m}, \mathbf{n}) = \mathbf{O}(\mathbf{g}(\mathbf{m}, \mathbf{n}))$ *if* $\exists\ C > 0, N > 0$ *such that* $m \geq N$ *and* $n \geq N \implies f(m, n) \leq Cg(m, n)$.

The proofs for the scenarios in problem 2.1 require you to make suitable reference to these definitions both to interpret the givens of each scenario and to complete the proof of the conclusion of the scenario. As it happens, once you have figured out the right C and N to use, proofs using the definitions of O and Ω are quite mechanical, and should look something like this:

- Let $C = ??$ and let $N = ??$
- Suppose $n \geq N$
- Then $f(n) ... \leq ... Cg(n)$

The reason to start with "let $C = ??$" and end with "$f(n) \leq Cg(n)$" is to make sure your argument starts with assumptions and flows towards conclusions. In many cases you will work backwards while solving the problem. For example, in part (a) below, you might ask yourself what C you need, and fool around with algebra until you determine, say, that $c = 6003$ makes everything work out great. That is, you start at your desired conclusion and work backwards towards the assumptions. This is a very powerful problem-solving technique, but its validity depends on your logical steps being reversible. And the very best way to show that your steps are reversible is to use them in a proof that explicitly flows from assumptions towards conclusions, and not the other way around.

So, I have taken pains below to ensure that every solution ends with that same visible and logical structure, highlighted in blue:

define your C and N and then show that whenever $n \geq N$, the required inequality holds true.

Problem 2.1(a)

Problem: *Define* $f : \mathbb{N} \to \mathbb{R}^+$ *by* $f(n) = 3n^2 + 1000n + 5000$*. Show that* $f(n) = O(n^2)$

Proof. Let $C = 6003$ and $N = 1$. Whenever $n \geq 1$, $n \leq n^2$. Therefore:

$$n \geq N \implies f(n) = 3n^2 + 1000n + 5000 \leq 3n^2 + 1000n^2 + 5000n^2 = 6003n^2 = Cn^2$$

Thus, $f(n) = O(n^2)$.

□

Problem 2.1(b)

Problems (b), (c), and (d) are slightly harder, because you need to use definitions 0.1-3 to interpret the assumptions of the problem, and then use those assumptions to construct a definition-based argument like in 2.1(a).

Suppose $g : \mathbb{N} \to \mathbb{R}^+$ *satisfies* $g(n) = \Omega(n^2)$*. Show that* $g(n) = \Omega(n)$

Proof. Since $g(n) = \Omega(n^2)$ we know that $\exists\ C > 0, N > 0$ such that $n \geq N \implies g(n) \geq Cn^2$.

Let $C' = C$ and let $N' = max(N, 1)$. Whenever $n \geq 1$, $n^2 \geq n$. Therefore:

$$n \geq N \implies g(n) \geq Cn^2 = C'n^2 \geq C'n$$

Thus, $g(n) = \Omega(n)$.

□

Problem 2.1(c)

Suppose $f, g : \mathbb{N} \to \mathbb{R}^+$ *satisfy* $f(n) = O(n)$ *and* $g(n) = O(n)$*. Let* $h(m, n) = f(m) + g(n)$*. Show that* $h(m, n) = O(m + n)$

Proof. Since $f(n) = O(n)$ we know that $\exists\ C_f > 0, N_f > 0$ such that $n \geq N_f \implies f(n) \geq C_f n$.

Similarly, since $g(n) = O(n)$ we know that $\exists\ C_g > 0, N_g > 0$ such that $n \geq N_g \implies g(n) \geq C_g n$.

Let $C = max(C_f, C_g)$ and let $N = max(N_f, N_g)$. Then:

$$m, n \geq N \implies h(m, n) = f(m) + g(n) \leq C_f m + C_g n \leq Cm + Cn = C(m + n)$$

Thus, $h(m, n) = O(m + n)$.

□

Problem 2.1(d)

Suppose $f, g : \mathbb{N} \to \mathbb{R}^+$ *and* $h : \mathbb{N} \times \mathbb{N} \to \mathbb{R}^+$ *is defined by* $h(m,n) = f(m) + g(n)$. *Show that if* $h(m,n) = O(m+n)$, *then both* $f(n) = O(n)$ *and* $g(n) = O(n)$.

Proof. Since $h(m,n) = O(m+n)$ we know that $\exists\, C > 0, N > 0$ such that $m, n \geq N \implies h(m,n) \leq C(m+n)$.

Let $C' = 2C$ and use the same N as above. Then:

$$n \geq N \implies f(n) \leq f(n) + g(N) = h(n,N) \leq C(n+N) \leq C(n+n) = 2Cn = C'n$$

Therefore, $f(n) = O(n)$.

Repeating this argument with f and g swapping places shows that $g(n) = O(n)$. □