

COMP 251 - Fall 2017- Assignment 1

Ian Benlolo
260744397
McGill University
October 4, 2017

1. (a) Let m_1, m_2, m_3, m_4 and w_1, w_2, w_3, w_4 be the four men and women being "matched". The following would be their list of preferences (from most preferred to least):

$m_1 : w_1, w_2, w_3, w_4$	$w_1 : m_2, m_4, m_3, m_1$
$m_2 : w_2, w_3, w_1, w_4$	$w_2 : m_3, m_4, m_1, m_2$
$m_3 : w_3, w_4, w_1, w_2$	$w_3 : m_1, m_4, m_2, m_3$
$m_4 : w_4, w_3, w_1, w_2$	$w_4 : m_1, m_3, m_2, m_4$

If this were the case, every man would be matched with their preferred woman but every woman would be stuck with their least preferred man since no other man is proposing to them after the first.

- (b) No, it is not possible to have a pairing where every man is paired with his least preferred woman. As a counter example lets say there are four people; two men and two women. First, m_1 will propose to his preferred woman (w_1) and she would say yes because he is first to propose to her. Now say m_2 proposes to the same woman, she will match with him if he is first on her list and wont if he is second on it. In the first case both women will therefore be matched with their favourite man but m_2 will also be matched with his favourite woman therefore we have not achieved the arrangement in question. In the latter case, m_2 will end up matched to w_2 , whom is second on his list but m_1 will remain matched to w_1 therefore again not achieving the arrangement in question. This is all assuming no two people have the same favourite other person (as seen in class). This logic can be extended to n men and n women.
- (c) No, this would not always lead to a stable matching because it can lead to either an infinite loop or an unstable matching. Take the following example:

$w_1 : w_3, w_2, w_4$	$w_3 : w_2, w_1, w_4$
$w_2 : w_1, w_3, w_4$	$w_4 : w_1, w_3, w_2$

In this case we can have 3 matchings:

- $w_1 - w_3$ and $w_2 - w_4$ but this would lead to an instability between w_3 and w_2 .
- $w_1 - w_2$ and $w_3 - w_4$ but this would leave w_1 and w_3 unstable.
- $w_1 - w_4$ and $w_2 - w_3$ but this would leave w_1 and w_2 unstable.

It is now proven that this is not a stable matching.

2. (a) $\sqrt{n} + n\sqrt{n} \leq n + n * n \leq n^2 + n^2 = 2n^2 \forall n \geq 1$. Therefore $\sqrt{n} + n\sqrt{n}$ is, in fact $O(n^2)$ with $c = 2$ and $n_o = 1$.
- (b) $(n + \log_2 n)^5 \leq (n + n)^5 = 2^5 n^5$ (for $n_0 \geq 1$). This proves $O(n^5)$ with $c = 2^5$ and $n_o = 1$. Now to show $(n^5) = O((n + \log_2 n)^5)$. $n^5 \leq n^5 + \log_2 n \leq (n^5 + \log_2 n)^5$ (for $n \geq 1$) $\therefore (n + \log_2 n)^5 = \Theta(n^5)$.
- (c)

$$\lim_{n \rightarrow \infty} \frac{n!}{n^n} = 0$$

since n^n grows much quicker than $n!$ therefore $n! = o(n^n)$.

- (d)

$$\lim_{n \rightarrow \infty} \frac{\log_2 n}{n^{\frac{1}{100}}} = \frac{\infty}{\infty} \stackrel{\text{HR}}{=} \dots \lim_{n \rightarrow \infty} \frac{100 \ln 2}{n^{\frac{1}{100}}} = 0$$

Therefore $\log_2 n$ is $o(n^{\frac{1}{100}})$.

3. (a) $2^{2^{n+1}} = (2^{2^n})^2 = (2^{2^n}) * (2^{2^n})$ which is not $\leq c * 2^{2^n} \therefore 2^{2^{n+1}} \neq O(2^{2^n})$.
 (b) $(\log_2 n)^5 \leq (5 \log_2 n)^5 = (\log_2 n^5)^5$ which is never less than $c * (\log_2 n^5)$ for any $c \geq 0, n \geq 1$.
 (c) Let's first prove that $1 = O(n^{1/n})$.

$$1 \leq n, \forall n \geq 1$$

Taking the n^{th} root of both sides you get

$$1 \leq n^{1/n}$$

Now to prove that $n^{1/n} = O(1)$ we use the fact that $2^n n, \forall n$. Again, taking the n^{th} root of both sides we get

$$2 > n^{1/n}$$

. Therefore $n^{1/n}$ is, in fact, $\Theta(1)$ with $c = 2$ and $n_0 = 0$.

- (d) $2^{(\log_2 n)^2} = O(n^{100})$. Taking the log of both sides we get

$$\log_2 (2^{(\log_2 n)^2}) = (\log_2 n)^2 \log_2 2 = 100 \log_2 n$$

Dividing both sides by $\log_2 n$ we get $\log_2 n * \log_2 2$ and 100. The statement

$$\log_2 n = 100$$

is false, and by taking the 2^{nd} power of both sides you can see that

$$2^{(\log_2 n)^2} \neq n^{100}$$

4. • This statement is false because the rule of sums for Big Oh states that $O(f+g) = \max\{O(f), O(g)\}$ and not $f(n) + O(g(n))$.

Proof. Let:

- $a_1(n) \leq c_1 f_1(n)$ for $n > n_1$, because $a_1 \in O(f_1)$.
- $a_2(n) \leq c_2 f_2(n)$ for $n > n_2$, because $a_2 \in O(f_2)$.

Summing these two inequalities we get:

$$a_1(n) + a_2(n) \leq c_1 f_1(n) + c_2 f_2(n) \leq \max\{c_1, c_2\} (f_1(n) + f_2(n)) \leq 2 * \max\{c_1, c_2\} * \max\{f_1(n), f_2(n)\}$$

QED

- This statement is also false because the Big Oh rule of products which states that $O(f * g) = O(f) * O(g)$ and not $f(n) * O(g(n))$.

Proof. Let:

- $a_1(n) \leq c_1 f_1(n)$ for $n > n_1$, because $a_1 \in O(f_1)$.
- $a_2(n) \leq c_2 f_2(n)$ for $n > n_2$, because $a_2 \in O(f_2)$.
- Multiplying these inequalities leads to the following:

$$a_1(n) a_2(n) \leq c_1 c_2 f_1(n) f_2(n)$$

for

$$n > \max\{n_1, n_2\}$$

With $\max\{n_1, n_2\}$ being n_0 .

QED

5. The base case was proved correctly. The fallacy is in the step where they equate n^2 with $O(n)$. Writing $n^2 = O(n)$ is not actually correct. What should really be written is $n^2 \in O(n)$ because $O(n)$ is really a set of functions by definition and not a number. This proof is therefore false because n^2 is not just a number and though the equal sign is there, it doesn't mean both sides are equal; it shows that the left side is an element of the right.