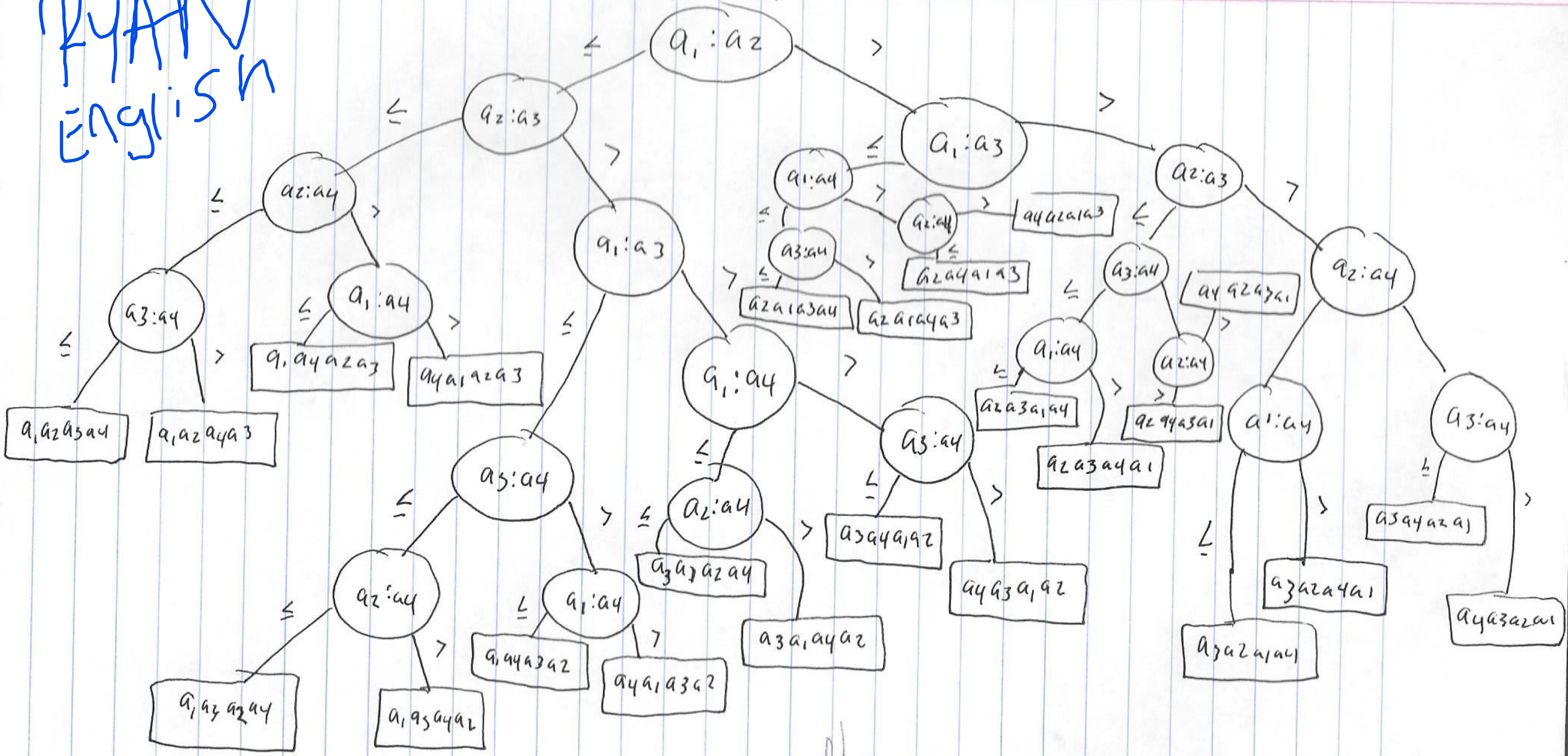
RYAN English

$A = [a_1, a_2, a_3, a_4]$



RYAN English

HW #2 3.1-2, 3.1-3, 3.1-4

3.1-2

Show that for any real constants a and b, where $b > 0$

$(n+a)^b = \Theta(n^b)$

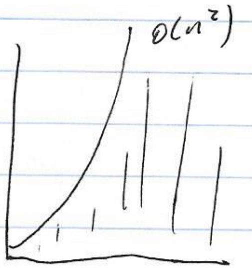
Expand the binomial

$C_0 n^{b-0} a^0 + C_1 n^{b-1} a^1 + C_2 n^{b-2} a^2 \ldots + C_b n^0 a^b$

$\hookrightarrow \le (C_0 + C_1 + C_2 \ldots + C_b) n^b = 2^b n^b$

3.1-3

Explain why the statement, "The running time of algorithm A is at least $O(n^2)$" is meaningless.



~~$T(n) \leq O(n^2)$~~

$T(n) \geq O(n^2)$

Using O asymptotic notation for a lower bound does not offer any information on the time complexity

3.1-4

Is $2^{n+1} = O(2^n)$? Is $2^{2n} = O(2^n)$

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

Given $2^{n+1} = 2 \times 2^n$ we know that $c \geq 2$ such that $2^{n+1} \leq c * 2^n$ therefore $2^{n+1} = O(2^n)$

$2^{2n} = O(2^n)$

Given $2^{2n} = 2^n * 2^n$ there is no value of c that makes $2^n * 2^n \leq c * 2^n$ therefore $2^{2n} \neq O(2^n)$