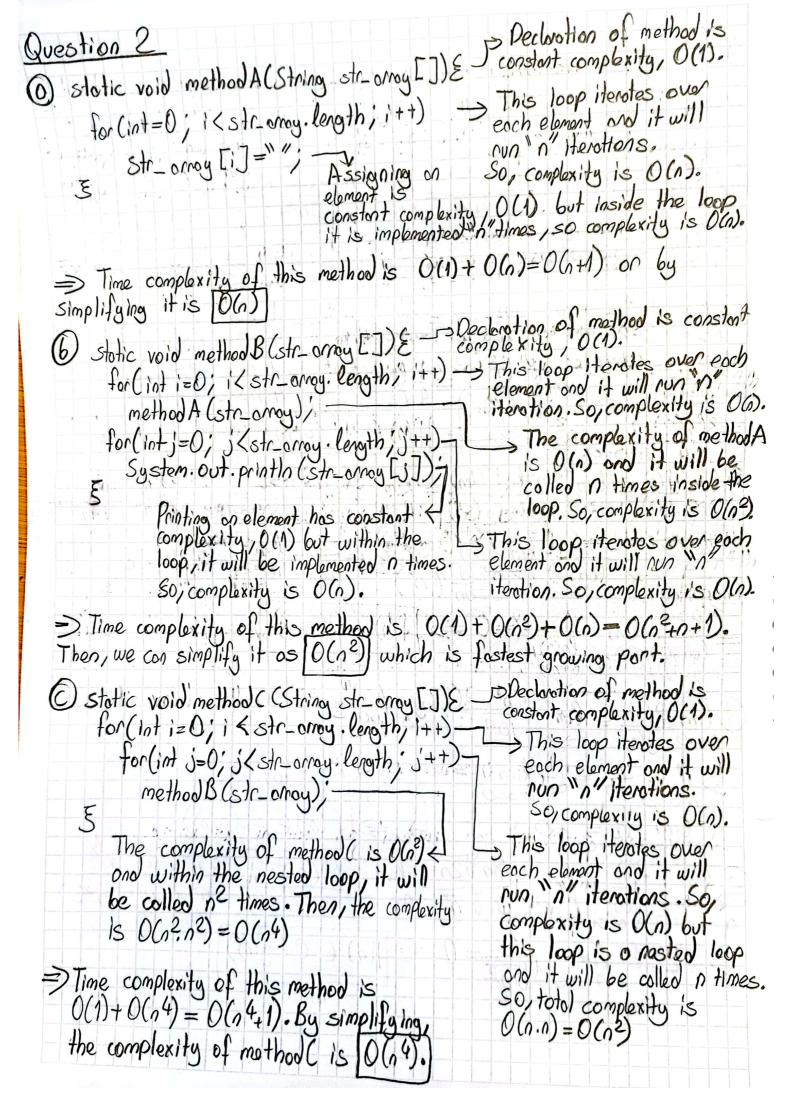
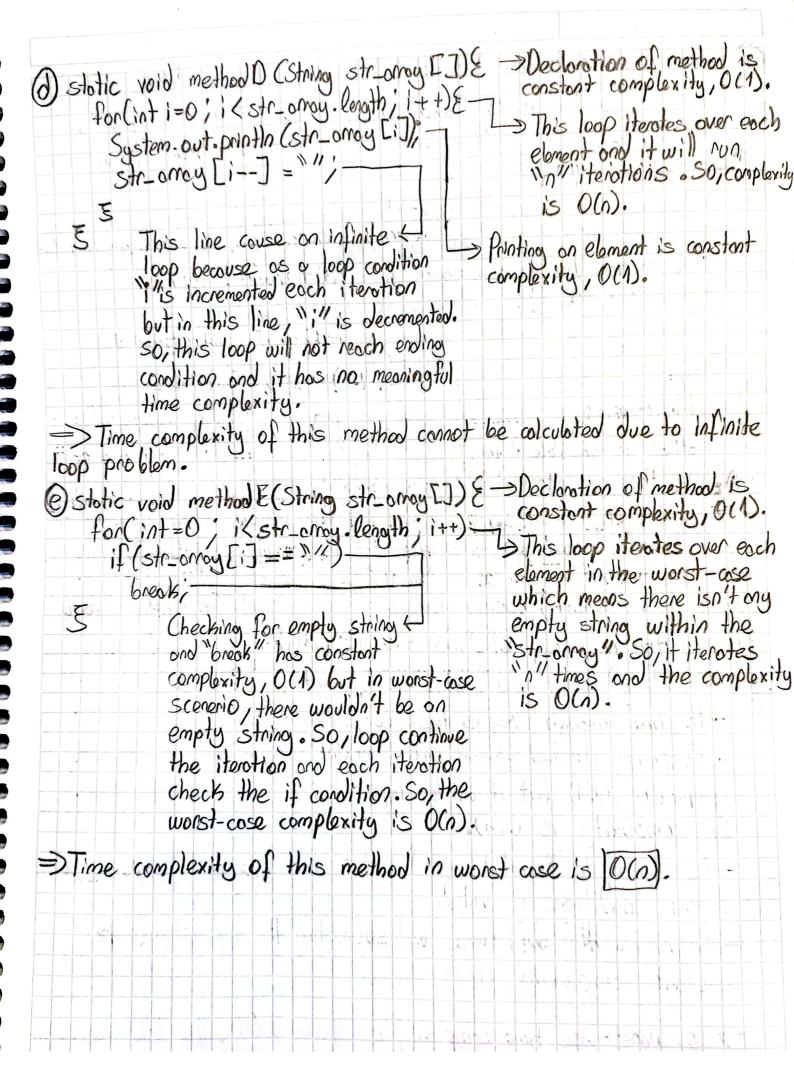
Data Structures and Algorithms-Homework #2 Nome / Surname ! Emre Kibor Student Number! 210104004093  $\frac{(\sqrt{vestion 1})}{0) \lim_{n \to \infty} \frac{f(n)}{g(n)} = \lim_{n \to \infty} \frac{(n^2 - 3n)^2}{sn^3 + n} = \lim_{n \to \infty} \frac{n^4 - 6n^3 + 3n^2}{sn^3 + n} = \lim_{n \to \infty} \frac{n^3 (n - 6 + \frac{a}{n})}{n^3 (s + \frac{1}{n^2})}$  $= \lim_{n \to \infty} \frac{n-6+\frac{q}{n}}{5+\frac{1}{n^2}} = \frac{\infty-6+\frac{q}{\infty}}{5} = \infty = \int f(n)\xi \Omega \left(g(n)\right)$   $= \int \lim_{n \to \infty} \frac{n-6+\frac{q}{n}}{5+\frac{1}{n^2}} = \lim_{n \to \infty} \frac{n^3}{5+\frac{1}{n^2}} = \lim_{n \to \infty} \frac{n^3}{6g_2n} = \lim_{n \to \infty}$ Using 1 lim & (n3) = 1 lim & 30 ln(2)

The spital rule 4 n >00 0 (logen) = 4 n >00 ln(2).0 4 n >00  $= \frac{3\ln(2) \lim_{n\to\infty} n^3 = \infty \Rightarrow f(n) \in \Omega(g(n))$  $O\lim_{n\to\infty}\frac{f(n)}{g(n)}=\lim_{n\to\infty}\frac{5n\cdot\log_2(4n)}{n\cdot\log_2(5n)}=\lim_{n\to\infty}\frac{5n\log_2(4n)}{n^2\log_2(5)}=\frac{5}{\log(5)}\lim_{n\to\infty}\frac{\log(4n)}{n}=\frac{5}{\log(5)}$ Using S lim on (loge(4n)) = S lim encon

Thospital rule loge(5) n>00 on (n) = loge(5) n>00 1  $=\frac{S}{\log_2(\omega)}\lim_{n\to\infty}\frac{1}{\ln(2)n}=0\Rightarrow \int_{\mathbb{R}} \int_$ =>  $\sqrt{2.8} \lim_{n\to\infty} \frac{1}{\sqrt{5\sqrt{n^2}}} = 0 \Rightarrow f(n) \in O(g(n))$ 





Quastion 3 METHOD Find Mox Difference (A)

IF A. length <2

RETURN ENDIFIE CONTRACTOR RETURN A[A.length-1]-A[O] ENDMETHOD · Method declaration has constant complexity, O(1).
· Checking the length of All for validity using it statement has · Returning the difference and subnecting two integers has both constant complexity, 0(2.1) = 0(1) The time complexity of this method is O(1) as the dominant factor in the worst-case scenario. (6) METHOD Find Max Difference (A) IF A. length <2 RETURN ENDIF moxInt = A[O] min lat= A [O] FOR i from 0 to A.length-1 IF A[i] > moxlot mox Int = A[i]ELSE IF A [i] < minInt min lot = A [i] ENDIF ENDFOR RETURN moxInt - minInt ENDMETHOD

· Declaration of the method has constant complexity, 0(1).
· Checking the length of "A" for validity using if statement has constant complexity, 0(1).

· Initializing "maxInt" and "minInt" as "A[0]" is constant complexity,

()(1).

The for loop that is used for finding the greatest and the smallest integer iterates over every element within the "A" and it has linear complexity , O(n).

· The if statements which controls the "A[i]" for "maxlet" and "minlet" have constant complexity, O(1) but within the loop, the · Returning the difference and subrocting "maxint" and "minint" has both

constant complexity, 0(1).

=> The time complexity of this method is O(1) as the dominant factor in the worst case scenario.