Süleyman Gölbol
1801042656

1) if $\lim_{n\to\infty} \frac{f(n)}{g(n)} = \infty$, then growth rate of $f(n)$ is larger than growth rate of $g(n)$.

if $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$, then growth rate of $f(n)$ is smaller than growth rate of $g(n)$. [It means $f(n) \in O(g(n))$]

$$T_2 < T_1 < T_4 < T_5 < T_3 < T_8 < T_6 < T_7$$

New Indices after Sorting: $4\log(\log n)$, $3\log n+3$, $2000n+1$, $(\frac{n}{6})^2$, n^5+8n^4, 2^n+n^3, 3^n+n^2, $n^n+1000n$

$\rightarrow \lim_{n\to\infty} \frac{4\log(\log n)}{3\log n+3} = \frac{\infty}{\infty}$ L'Hospital $\Rightarrow \frac{\frac{4}{n\log n}}{\frac{3}{n}} = \frac{4n}{3n\log n} = \frac{4}{3}\lim_{n\to\infty} \frac{1}{\log n} = \frac{4}{3}.0 = 0$

$T_2(n) \in O(T_1(n))$

$\rightarrow \lim_{n\to\infty} \frac{3\log n+3}{2000n+1} = \frac{\infty}{\infty}$ L'Hospital $\Rightarrow \frac{\frac{3}{n}}{2000} \Rightarrow \frac{3}{2000}\lim_{n\to\infty}\frac{1}{n} = \frac{3}{2000}.0=0$ $T_1(n) \in O(T_4(n))$

$\rightarrow \lim_{n\to\infty} \frac{2000n+1}{(\frac{n}{6})^2} = \frac{36.(2000n+1)}{n^2} \Rightarrow 72000.\lim_{n\to\infty}\frac{1}{n} + \lim_{n\to\infty}\frac{1}{n^2} = 72000.0+0=0$

$T_4(n) \in O(T_5(n))$

$\rightarrow \lim_{n\to\infty} \frac{(\frac{n}{6})^2}{n^5+8n^4} \Rightarrow \frac{1}{36}\lim_{n\to\infty}\frac{n^2}{n^5+8n^4} \overset{\text{L'Hospital}}{=} \frac{\infty}{\infty} \Rightarrow \frac{1}{36}\lim_{n\to\infty}\frac{2n}{5n^4+32n^3} \overset{\text{L'Hospital}}{=} \frac{\infty}{\infty} = \frac{1}{36}\lim_{n\to\infty}\frac{2}{20n^3+96n^2} = \frac{1}{36}.0=0$

$T_5(n) \in O(T_3(n))$

$\rightarrow \lim_{n\to\infty} \frac{n^5+8n^4}{2^n+n^3} \overset{\text{L'Hospital}}{=} \frac{\infty}{\infty} \Rightarrow \frac{5n^4+32n^3}{2^n.\log 2+3n^2} \overset{\text{L'Hospital}}{=} \frac{\infty}{\infty} \Rightarrow \frac{20n^3+96n^2}{2^n.(\log 2)^2+6n} = \frac{\infty}{\infty}$ The power of nominator decreases but power of denominator doesn't change because of 2^n.

At the end $\lim_{n\to\infty} \frac{\text{constant}}{2^n.\text{constant}} \Rightarrow \lim_{n\to\infty}\frac{1}{2^n} = 0$ $T_3(n) \in O(T_8(n))$

$\rightarrow \lim_{n\to\infty} \frac{2^n+n^3}{3^n+n^2} \overset{\text{L'Hospital}}{=} \frac{\infty}{\infty} \Rightarrow \frac{2^n\log 2+3n^2}{3^n\log 3+2n} \overset{\text{L'Hospital}}{=} \frac{\infty}{\infty} = \frac{2^n(\log 2)^2+6n}{3^n(\log 3)^2+2} \overset{\text{L'Hospital}}{=} \frac{\infty}{\infty} = \frac{2^n.(\log 2)^3+6}{3^n(\log 3)^3}$

At the end $\frac{\text{Constant 1}}{\text{Constant 2}}.\lim_{n\to\infty}\frac{2^n}{3^n} \Rightarrow \lim_{n\to\infty}(\frac{2}{3})^n = 0$ $T_8(n) \in O(T_6(n))$

$\rightarrow \lim_{n\to\infty} \frac{3^n+n^2}{n^n+1000n}$ = Divide by n^n = $\frac{\lim_{n\to\infty}\frac{3^n}{n^n} + \lim_{n\to\infty} n^{2-n}}{\lim_{n\to\infty}(1+1000n^{1-n})} = \frac{0+0}{1+0} = 0$

$T_6(n) \in O(T_7(n))$

$T_i(n) \in O(T_j(n))$ for every $T_i < T_j$ where $1 \leq i,j \leq 8$

Süleyman Gölbol
1801042656

2) $\lim_{n\to\infty} \frac{f(n)}{g(n)} = L \begin{cases} \text{if } L=0, & f(n) \in O(g(n)) \\ \text{if } L=c, & f(n) \in \Theta(g(n)) \\ \text{if } L=\infty, & f(n) \in \Omega(g(n)) \end{cases}$

a) $\lim_{n\to\infty} \frac{99n}{n} = 99$ (constant) so $f(n) \in \Theta(g(n))$

b) $\lim_{n\to\infty} \frac{2n^4+n^2}{(\log n)^6} = \frac{\infty}{\infty}$ so L'Hospital. $\lim_{n\to\infty} \frac{8n^3+2n}{\frac{6(\log n)^5}{n}} = \frac{n(8n^3+2n)}{6(\log n)^5}$

$= \frac{\infty}{\infty} \xRightarrow{\text{Lopital}} \frac{32n^3+4n}{\frac{30.(\log n)^4}{n}} = \lim_{n\to\infty} \frac{32n^4+4n^2}{30(\log n)^4} = \frac{\infty}{\infty} \xrightarrow{\text{Lopital}}$

As it seen, the power of numerator doesn't change, but power of denominator decreases. So at the end it will be $\frac{c.\infty}{\text{Constant}} = \infty$

So $f(n) \in \Omega(g(n))$

$(2n^4+n^2) \in \Omega((\log n)^6)$

c) $\lim_{n\to\infty} \frac{\sum_{x=1}^{n} x}{4n+\log n} = \lim_{n\to\infty} \frac{n(n+1)}{2(4n+\log n)} = \frac{1}{2} \lim_{n\to\infty} \frac{n+1}{\frac{\log n}{n}+4}$

$\lim_{n\to\infty} \underbrace{\frac{\log n}{n}}_{\frac{1}{n^2} \text{ with L'Hospital}} = 0$ so $\frac{1}{2}\lim_{n\to\infty}(n+1) = \frac{\infty}{8} = \infty$ $\quad \sum_{x=1}^{n} x \in \Omega(4n+\log n)$

d) $\lim_{n\to\infty} \frac{3^n}{5^{\sqrt{n}}} = \lim_{n\to\infty} 3^n . 5^{-\sqrt{n}} \Rightarrow \lim_{n\to\infty} e^{n.\log 3 - \sqrt{n}.\log 5}$

$\boxed{\text{Logarithm-exponential conversion Rule} \quad e^{\log_e x} = x}$

$= e^{\lim_{n\to\infty}(\sqrt{n}(\log 3.\sqrt{n} - \log 5))}$

$= e^{\underbrace{\lim_{n\to\infty}(\sqrt{n})}_{\infty} . \lim_{n\to\infty}(\log 3 . \sqrt{n} - \log 5)} = e^{\infty(\log 3 . \underbrace{\lim_{n\to\infty}\sqrt{n}}_{\infty} - \log 5)}$

$= e^{\infty.(\log 3 . \infty - \log 5)} = e^{\infty . \infty} = e^{\infty} = \infty$

$3^n \in \Omega(5^{\sqrt{n}})$

Süleyman [illegible]
1801042656

3) a) This function takes 2 arguments; an array and size n.
If at least half of array contains same number, it returns that number; else, it returns −1.
Outer loop for iterate over array with index i. Inner loop is also for iterate over array with index j. And count, counts the number of repeating.

b) If count exceeds $\frac{n}{2}$ in the last iteration of loop or if it doesn't exceed $\frac{n}{2}$, then it means we will iterate all the loop.
Because of nested loops, in the worst case, this means $O(n.n)=O(n^2)$
In the best case, most repeated element is first element.
So, outer for loop will not be iterated again. Inner loop will be iterated at least $\frac{n}{2}$. $\Omega(\underbrace{1}_{\text{constant}}+\frac{n}{2}) = \Omega(n)$ Best case

4) a) Nums is array and n is size of array. First we find the biggest number in array. Then we create a dynamic array with calloc.
If array contains a number, index with that number will not be 0.
Return value is the number if it's bigger than half of size.
And if doesn't exist, it will return −1.
Map's indexes are the numbers in array and it contains the repetition time.

b) Time complexity
In the worst case, we will iterate all three for loops.
$O(3.n) = O(n)$
In the best case even if we don't iterate in the last for loop, we have to iterate on the first 2. So $\Omega(n)$.

5) For the time complexity; in the best case 3 and 4 are both $O(n)$.
But for the worst case, algorithm in 4 is better.
For the space complexity,
For 3, each iteration of loop will need $O(1)$ space for temporaries, but when loop finishes it can be resused so total space needed $O(1)/\Omega(1)$.
For 4, because of allocating with calloc and not freeing; it is $\overset{\text{worst}}{O(n)} / \overset{\text{best}}{\Omega(n)}$.

Süleyman Gölbol 1801042656

6)

a)

```
max = 0 , i = 1, j = 1
while i ≤ n:
    while j ≤ m:
        if A[i] * B[j] > max
            max = A[i] * B[j]
    end while
end while
```

In the worst and best case, we have to iterate completely on both so complexity will be $O(n^2)$ and $\Omega(n^2)$

b) Note: These are not real array indices, They start with 0.

```
calloc(allocate) n+m memory to mergedArray
i = 1
while i ≤ n:
    mergedArray[i] = A[i]
    i++
j = 1
while j ≤ m:
    mergedArray[n+j] = B[j]
    j++
i = 1
while i ≤ n+m:
    j = i+1
    while j ≤ n+m:
        if mergedArray[i] < mergedArray[j]:
            swap mergedArray[i] and mergedArray[j]
        j++
```

First 2 loops are linear and third one is quadratic. In any case, we have to iterate on all merged array so $O(n^2+n+n) \Rightarrow O(n^2)$ $\Omega(n^2+n+n) \Rightarrow \Omega(n^2)$

c)

```
calloc to allocate n+1 sized memory to newArray
i = 1
while i ≤ n:
    newArray[i] = array[i]
newArray[n] = elementToBeAdded
```

Because of one loop, it will take $O(n)$ in worst case. If we presume that we don't have allocated empty slot; also best case is $\Omega(n)$.

d)

```
i = 1
while i ≤ n:
    if array[i] == elementToDelete:
        j = i-1
        while j ≤ n-1:    # shifting
            array[j] = array[j+1]
        break
```

If we delete the first element of array, we will not iterate on outer loop so complexity will be $\Omega(n)$ in best case. Also if element doesn't exist we will not use inner loop so $\Omega(n)$. In the worst case, we will find the element but not first so complexity will be $O(n*n) = O(n^2)$