

2022, Fall

학과: 정보컴퓨터공학부

학번: 201924437

이름: 강근하

HOMEWORK # 2

[분석과 리스트]

1. The number of operations executed by algorithms A and B is $8n \log n$ and $2n^2$, respectively. Determine n_0 such that A is better than B for $n \geq n_0$.

(Text book Exercise No: R-4.7)

A. A가 B보다 더 나은 알고리즘 \Rightarrow A의 수행 시간 \leq B의 수행 시간 이어야 한다.

따라서 $8n \log n \leq 2n^2$, $4 \log n \leq n$ 이 된다. (임계값 n 의 크기는 1 이상)

$O(n \log n) < O(n^2)$ 이 항상 성립함을 알고 있으므로, 위 식은 $n=1$ 이상일 때 항상 성립하게 된다. (최소시간 < 평방시간). 구하고자 하는 $n_0 = 1$ 이 된다.

답: $n_0 = 1$

2. What is the sum of all the even numbers, $E(n)$ from 0 to $2n$, for any positive integer n ? (Text book exercise No: R-4.11)

A. n 항까지의 $0 \sim 2n$ 에서의 합 (짝수만)을 구해야 한다.

$$n=1 : 0 + 2 = 2$$

$$n=2 : 0 + 2 + 4 = 6$$

$$n=3 : 0 + 2 + 4 + 6 = 12$$

$$\vdots$$

$$n=n : 0 + 2 + 4 + 6 + \dots + 2n = \frac{(2 + 2n) \cdot n}{2} = n(n+1) = n^2 + n.$$

따라서 구하고자 하는 $E(n) = n^2 + n$ 이다.

답: $n^2 + n$

3. Order the following functions by asymptotic growth rate. 정렬!!

(Text book exercise No : R-4.13)

$$\left. \begin{array}{lll} \textcircled{1} 4n \log n + 2n & \textcircled{2} 2^{10} & \textcircled{3} 2^{\log n} \\ \textcircled{4} 3n + 100 \log n & \textcircled{5} 4n & \textcircled{6} 2^n \\ \textcircled{7} n^2 + 10n & \textcircled{8} n^3 & \textcircled{9} n \log n \end{array} \right\}$$

A. 위의 주어진 식들을 수업시간에 배운 notation 으로 새롭게 표현하자. (최고차항을 따르도록)

① $4n \log n + 2n$

$\Rightarrow n \log n$ 과 n 을 비교하면, $n \log n$ 이 더 큰 항이 된다. $\Rightarrow \Theta(n \log n)$

② 2^{10}

\Rightarrow constant time 이므로 $\Theta(1)$ 이다.

③ $2^{\log n}$

\Rightarrow 그대로 $\Theta(2^{\log n})$ 이다.

④ $3n + 100 \log n$

$\Rightarrow n$ 과 $\log n$ 을 비교하면, $O(n) > O(\log n)$ 이므로 $\Theta(n)$ 이 된다.

⑤ $4n$

$\Rightarrow n$ 그대로 $\Theta(n)$ 이다.

⑥ 2^n

\Rightarrow 그대로 $\Theta(2^n)$ 이다.

⑦ $n^2 + 10n$

$\Rightarrow n^2$ 이 최고차항 이므로 $\Theta(n^2)$ 이다.

⑧ n^3

\Rightarrow 그대로 $\Theta(n^3)$ 을 따른다.

⑨ $n \log n$

\Rightarrow 그대로 $\Theta(n \log n)$ 이 된다.

위 결과에서, 같은 값으로 나온 친구들끼리 묶으면 ①과 ④, ⑤와 ⑧가 있는데 이들은 최고차항의 계수를 비교해주면 된다. 따라서 연산 시간의 크기를 순서대로 정렬하면,

$$\Theta(1) < \Theta(2^{\log n}) < \Theta(n) < \Theta(n \log n) < \Theta(n^2) < \Theta(n^3) < \Theta(2^n)$$

$$\downarrow \quad \quad \quad \downarrow \quad \quad \quad \downarrow \quad \quad \quad \downarrow \quad \quad \quad \downarrow \quad \quad \quad \downarrow \quad \quad \quad \downarrow$$

$$2 < 3 < 4 < 5 < 9 < 7 < 1 < 8 < 6$$

따라서, 구하고자 하는 순서는

$$2^{10} < 2^{\log n} < 3n + 100 \log n < 4n < n \log n < 4n \log n + 2n < n^2 + 10n < n^3 < 2^n \text{ 이다.}$$

답

4. Give a log-oh characterization, in terms of n , of the running time of the Example function shown in the following Code Fragment. (Text Book Exercise No: R-4.20)

Algorithm Ex 5 (A, B):

Input : Arrays A and B each storing $n \geq 1$ integers.

Output : The number of elements in B equal to the sum of prefix sums in A.

1행	$c \leftarrow 0$	$\rightarrow 1$ 번
2행	for $i \leftarrow 0$ to $n-1$ do	$\rightarrow n+1$ 번
3행	$s \leftarrow 0$	$\rightarrow n$ 번
4행	for $j \leftarrow 0$ to $n-1$ do	$\rightarrow (n+1) \cdot (n+1)$
5행	$s \leftarrow s + A[0]$	$\rightarrow ?$
6행	for $k \leftarrow 1$ to j do 최악화항!!	$\rightarrow (n+1)(n+1) \dots$
7행	$s \leftarrow s + A[k]$	
8행	if $B[i] = s$ then	
9행	$c \leftarrow c + 1$	
10행	return c	

\Rightarrow 3개의 for 문이 존재하지만, 6행은 4행과 마찬가지로 실행되므로, $O(n^3)$ 이 된다.
(최악화항을 따지기 때문!)

답: $O(n^3)$

5. Show that 2^{n+1} is $O(2^n)$. (Text Book Exercise No: R-4.30)

A. $2^{n+1} = 2^n \times 2$ 이다.

따라서 2×2^n 안에, 빅 오 표기법에서 최악화의 계수는 무시함을 알 수 있다.

(n 이 무한대로 커질 시 사실상 계수가 주는 영향이 매우 적다)

따라서 빅 오 표기법 값을 대 $O(2^n)$ 이다.

6. Show that $n \log n$ is $\Omega(n)$. (Text Book Exercise No. R-4.33)

A. Big- Ω 표기법의 경우, Lower bound 중 최대인 것을 찾아야 한다.

앞서 배운 연산 시간의 크기 순서대로 따지면, $O(n) > O(n \log n)$ 이 되므로 원하는 greatest lower bound는 $\Omega(n)$ 이 된다.

시간 $n \log n > n \quad (n > 1)$

7. 리스트를 정의하고 이 자료구조를 위한 List ADT를 규정하십시오.

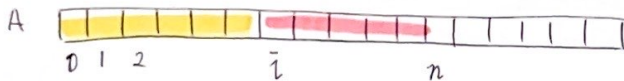
A. 리스트란, 자료들의 개수가 유한하며 순서가 있는 (정렬된) 구조이다. 각각의 요인에는 그의 위치가 있다.

List ADT 에서 ADT는 추상 자료형들을 말한다. 그는 리스트의 자료 구조화를 해 List ADT를 사용하면 이 기본적인 구조로 여러가지 자료구조를 만들어낼 수 있다.

8. 리스트를 배열로 구현할 때의 장점과 단점을 기술하십시오.

A. 리스트를 배열을 통해 구현하면, 순차적인 맵핑을 통해 캐시 메모리는 잘 활용하고 쉽게 구현이 가능한 반면에, insert와 delete를 실행할 때 오버헤드가 발생할 수도 있다는 단점도 있다.

9. 배열 A로 구현된 리스트에서 i 위치 object를 입력하는 메소드 $insert(i, obj)$ 를 구현하려고 한다. 이 메소드가 실행될 때의 과정을 설명하십시오.



A. 인서트 될 때, i 번째에 입력을 하려면 입력 하나 만큼 새로운 공간이 필요하다. 그래서 i 번째부터 한 칸씩 앞으로 이동하고, i 자리는 비워 거기서 object를 넣게 된다.

만약 i 번째가 배열의 첫 시작 주소라면, 이 행동을 n 번 해야 하므로 $O(n)$ 시간이다. 것이다.

