

알고리즘 Homework 1 (마감일: 4월 27일 월요일 오후 9시) 제출일: ____월 ____일 ____요일

성명 : _____ 학과: _____ 학번: _____

◎ 본 Homework을 수행하면서 다른 학생의 문서로부터 일부 또는 전체를 복사하였습니까?
예() 아니오() (복사하였으면 예에 체크하고 아니라면 아니오에 체크하시오)

다음 연습문제 16문항을 풀어, 4월 27일 월요일 오후 9시까지 아주Bb에 제출한다. 손으로 풀이한 것을 사진 찍거나 스캔하여 하나의 문서 파일로 묶어 제출하여도 좋다.

반드시 “Introduction to the Design & Analysis of Algorithms, 3rd Ed. (International Edition)”의 문제를 풀이하며, 다른 판본에 수록된 문제를 풀이하지 않도록 한다. 또한, 교재에서 정확하지 않게 기술된 문제 2번, 문제 4번, 문제 5번, 문제 8번과 문제 13번은 아래에 빨간색 글씨로 수정하니, **빨간색으로 수정된 부분을 기준으로 문제를 풀이**한다.

1. [교재 84~85쪽 Exercise 2.2 2번] (12점, 부분문제 (a~d) 각 3점)

부분문제 (a) ~ (d)를 푸시오.

sol)

$2n(n-1)/2 \in \Theta(n^2)$ 이므로

a) true b) true c) false d) true

2. [86쪽 Exercise 2.2 5번 - **빨간색 부분 변경**] (14점)

7개 수식 중, $0.05n^{10} + 3^{n^3} + 1$ 을 $0.05n^{10} + 3n^3 + 1$ 로 변경.

변경 전	변경 후
$0.05n^{10} + 3^{n^3} + 1$	$0.05n^{10} + 3n^3 + 1$

sol)

$2\log(n+50)^5 \in \Theta(\log n)$, $\ln^3 n \in \Theta(\log^3 n)$, $\sqrt{n} \in \Theta(n^{1/2})$, $0.05n^{10} + 3n^3 + 1 \in \Theta(n^{10})$,
 $3^{2n} \in \Theta(3^{2n})$, $3^{3n} \in \Theta(3^{3n})$, $(n^2 + 3)! \in \Theta(n^2!)$

3. [86쪽 Exercise 2.2 7번] (총 20점, 부분문제 (a~d) 각 5점)

부분문제 (a) ~ (d)를 푸시오.

sol)

a.

$t(n) \in O(g(n))$ 이면, $t(n) \leq c g(n)$ for all $n \geq n_0$ 가 성립한다 (단, $c > 0$)

위 가정에 따라, $g(n) \geq (1/c) t(n)$ for all $n \geq n_0$ 또한 성립하고,

Big-Omega의 정의에 따라 $g(n) \in \Omega(t(n))$ 이 성립한다.

b. 임의의 $t(n)$ 에 대하여, 이 $t(n) \in \Theta(g(n))$ 라고 가정하자.

$t(n) \in \Theta(g(n))$ 이면 Big-Theta의 정의에 따라, $c g(n) \leq t(n) \leq c g(n)$ 이 성립한다.

이 때, 양 변에 양수 α 를 곱하면, $\alpha c g(n) \leq t(n) \leq \alpha c g(n)$ 가 성립한다.

따라서, $\Theta(\alpha g(n)) = \Theta(g(n))$ 이 성립한다.

c. 어떤 $f(n)$ 에 대하여, $f(n) \in \Omega(g(n))$ 이면 $f(n) \geq cg(n)$ 이 성립한다.

또한, $f(n) \in O(g(n))$ 이면, $f(n) \leq cg(n)$ 이 성립한다.

이 때, 각각의 교집합은 $cg(n) \leq f(n) \leq cg(n)$ 이므로 Big-Theta 의 정의를 만족한다. 그러므로 명제가 성립한다.

d. counter example

$$f(n) = \begin{cases} n^3 & \text{if } n \text{ is even} \\ n & \text{if } n \text{ is odd} \end{cases} \quad g(n) = \begin{cases} n & \text{if } n \text{ is even} \\ n^3 & \text{if } n \text{ is odd} \end{cases}$$

n 이 증가하면서 짝수, 홀수가 반복됨에 따라서 $f(n)$ 과 $g(n)$ 의 증가율이 계속 뒤바뀜.

4. [94쪽 Exercise 2.3 5번-a~d - 빨간색 부분 변경] (총 14점, 부분문제 a,b,d 각 3점, c 5점)

아래 표의 코드 부분을 변경 후, 부분문제 (a) ~ (d)를 푸시오. [문제의 설명대로 이 문제 앞문제인 Exercises 2.3의 Problem 4에 있는 a ~ d를 풀면 됩니다. e는 풀 필요 없음.]

주어진 알고리즘 중 아래 빨간색 부분을 변경함.

변경 전	변경 후
...	...
if $A[i] > val$	if $A[i] > val$
$sumgreater \leftarrow A[i]$	$sumgreater \leftarrow sumgreater + A[i]$
if $A[i] < val$	if $A[i] < val$
$sumless \leftarrow A[i]$	$sumless \leftarrow sumless + A[i]$
...	...

sol)

a. A 배열의 원소들 중 100보다 큰 수들의 합과 100보다 작은 수들의 합의 차이를 구하는 알고리즘

b. val 값과 배열에 원소들과의 비교

c. $\sum_{k=0}^{n-1} 2 = 2n$

d. $\Theta(n)$

5. [102쪽 Exercise 2.4 1번 - 빨간색 부분 변경됨] (총 20점, 부분문제 (a~e) 각 4점)

다음 표의 수식부분을 변경한 후, 부분문제 (a)~(e)를 푸시오.

문제	변경 전	변경 후
a	$x(n) = 2x(n-3)$ for $n > 1$, $x(1) = 1$	$x(n) = 2x(n-1)$ for $n > 1$, $x(1) = 1$
b	$x(n) = x(n-2) - 2$ for $n > 1$, $x(1) = 0$	$x(n) = x(n-1) + 2$ for $n > 1$, $x(1) = 0$
c	$x(n) = 2x(n-2) - n$ for $n > 0$, $x(0) = 0$	$x(n) = 2x(n-1) + n$ for $n > 0$, $x(0) = 0$
d	$x(n) = x(n/2) - 2$ for $n > 1$, $x(1) = 2$ (solve for $n = 4^k$)	$x(n) = x(n/2) + 2$ for $n > 1$, $x(1) = 2$ (solve for $n = 2^k$)
e	변경 없음	변경 없음

sol)

a. $x(n)/x(n-1) = 2 \rightarrow$ 등비수열
 $\rightarrow x(n) = 2^{n-1} \cdot x(1)$
 $= 2^{n-1} \cdot 3$

b. $x(n) - x(n-1) = 2 \rightarrow$ 등차수열
 $\rightarrow x(n) = 2(n-1) + x(1)$
 $= 2(n-1)$

c. $x(n) - x(n-1) = n \rightarrow$ 계차수열
 $\rightarrow x(n) = \sum_{k=1}^n k + x(0)$
 $= n(n+1)/2$

d. $\leq t \ n = 2^k, x(2^k) = f(k)$
 $\rightarrow f(k) = f(k-1) + 2$
 $\rightarrow f(k) - f(k-1) = 2 \rightarrow$ 등차수열
 $\rightarrow f(k) = 2(k-1) + f(1)$
 $f(1) = x(2) = x(1) + 2 = 4$
 $\rightarrow f(k) = 2k + 2$
 $\rightarrow x(2^k) = 2k + 2, k = \log_2 n$
 $\rightarrow x(n) = 2\log_2 n + 2$

e. $\leq t \ n = 5^k, x(5^k) = f(k)$
 $\rightarrow f(k) = f(k-1) + 5^k$
 $\rightarrow f(k) - f(k-1) = 5^k$
 $\rightarrow f(k) = \sum_{i=1}^k 5^i + f(0), f(0) = x(1) = 1$
 $= \sum_{i=0}^k 5^i = (5^{k+1} - 1) / (5 - 1) = x(5^k)$
 $\rightarrow x(n) = (5n - 1) / 4$

6. [104쪽 Exercise 2.4 8번] (총 20점, 부분문제 (a~d) 각 5점)

부분문제 (a)~(d)를 푸시오.

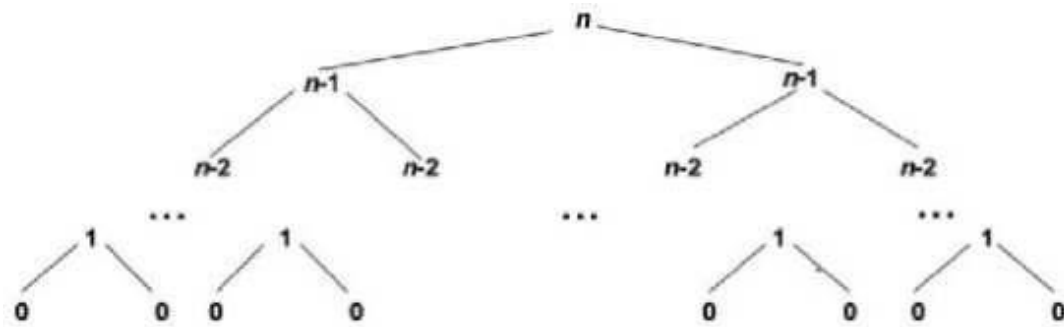
sol)

a. *Algorithm Pow(n)*
if $n = 0$ return 1
else return $Pow(n-1) + Pow(n-1)$

b. $A(n) = 2A(n-1) + 1, A(0) = 0$

$$\begin{aligned}
A(n) + 1 &= 2A(n-1) + 2, \quad A(n) + 1 = f(n) \\
\rightarrow f(n) &= 2f(n-1) \\
\rightarrow f(n)/f(n-1) &= 2 \\
\rightarrow f(n) &= 2^{n-1} \cdot f(1), \quad f(1) = A(1) + 1 = 2 \\
\rightarrow A(n) + 1 &= 2^n \\
\rightarrow A(n) &= 2^n - 1
\end{aligned}$$

c.



$$\text{number of calls} = \sum_{k=1}^n 2^k = 2^{n+1} - 2$$

d. 효율적이지 않은 알고리즘이다.

n의 크기가 커질수록 연산이 2^n 만큼 늘어나기 때문에 time complexity가 높고 단순히 2를 n회 곱하는 알고리즘보다 더 비효율적인 알고리즘이다.

7. [교재 128쪽 Exercise 3.1 8번] (10점)

sol)

```

| S O R T I N G
G | O R T I N S
G I | R T O N S
G I N | T O R S
G I N O | T R S
G I N O R | T S
G I N O R S | T

```

8. [교재 133쪽 Exercise 3.2 4번 - 빨간색 부분 변경] (10점)

변경 전	변경 후
... it is 43 characters long it is 44 characters long ...

sol)

42번 비교.

pattern을 text에 맞춰 놓고 비교를 시작하는 것이 전체 {(text 글자 수) - (pattern 글자 수) + 1}번 = 44-6+1=39번이다.

제일 먼저, SORTIN과 SEARCH를 맞춰 놓고 앞글자부터 비교를 하는데, 첫글자 S는 일치하고 O와 E는 일치하지 않으므로, 두 번의 비교만에 패턴의 위치를 한 칸 뒤로 옮겨 놓고 다시 비교를 한다. 마찬가지로 방법으로 할 때, 대부분 첫 문자 S가 text 내의 글자와 달라 한번의 비교만에 다음 텍스트 자리로 패턴을 옮기게 된다. 그 외에 한번 비교 만에 패턴이 이동하지 않는 경우는, ...(U)SE_BRU...에서 S, E 일치, 세 번째 글자 _와 A가 다른 상황이 되어 세 번의 비교가 일어나는 경우 뿐이다.

따라서, 전체적으로 $1*2 + 37*1 + 1*3 = 42$ 번의 문자 비교가 일어난다.

9. [146쪽 Exercise 3.4 1-a번] (10점)

부분문제 (a)를 푸시오. [부분문제 b는 풀 필요 없음.]

sol)

a. $O(n*2^n)$

모든 부분집합의 개수는 2^n 이다. basic operation을 weight와 value들의 합을 구하는 덧셈으로 하였을 때, 전체 덧셈의 횟수는 worst case $2n*2^n$ 번이다. 문제에서 각각의 부분집합을 구하는 시간이 constant하다고 했으므로, 이를 c라고 하면, 전체 time complexity는 $T(n) = c*2n*2^n \in O(n*2^n)$ 이다.

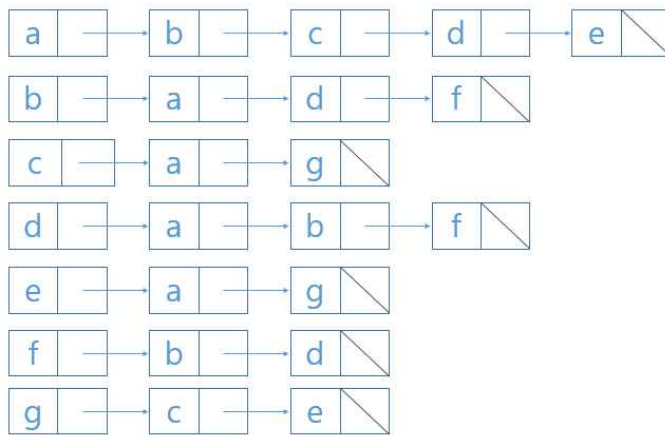
10. [154쪽 Exercise 3.5 1-a번] (총 10점, 부분문제 (a~b) 각 5점)

(부분문제 (b)에서 “resolving ties by the vertex alphabetical order” 는, 한 vertex에서 갈 수 있는 다음 vertex가 여러 개 있을 때, vertex에 있는 문자를 보고, 알파벳 순서에서 앞에 오는 것을 먼저 방문하라는 뜻임.)

<Sol>

a.

	a	b	c	d	e	f	g
a	0	1	1	1	1	0	0
b	1	0	0	1	0	1	0
c	1	0	0	0	0	0	1
d	1	1	0	0	0	1	0
e	1	0	0	0	0	0	1
f	0	1	0	1	0	0	0
g	0	0	1	0	1	0	0

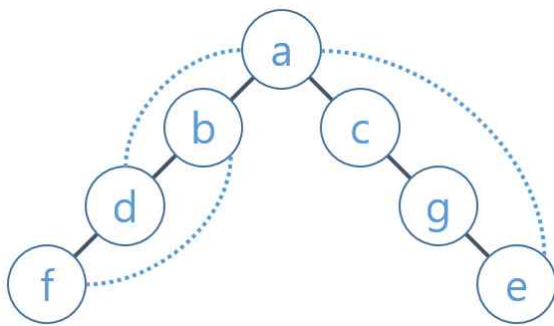


b.

push order : $a \rightarrow b \rightarrow d \rightarrow f \rightarrow c \rightarrow g \rightarrow e$

pop order : $f \rightarrow d \rightarrow b \rightarrow e \rightarrow g \rightarrow c \rightarrow a$

tree :



11. [교재 163쪽 Exercise 4.1 7번] (10점)

<sol>

C O N Q U E R

C | O

C O | N

C N O | Q

C N O Q | U

C N O Q U | E

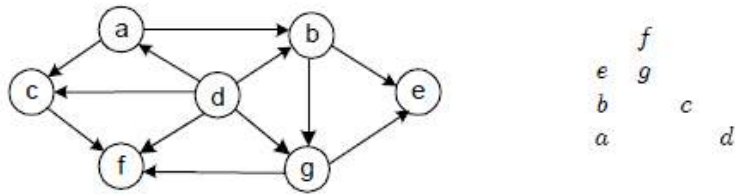
C E N O Q U | R

C E N O Q R U

12. [교재 168쪽 Exercise 4.2 1번] (5점)

그림 (a)에 대해서만 푸시오. (그림 (b)에 대해서는 풀 필요 없음.)

1. a. The digraph and the stack of its DFS traversal that starts at vertex a are given below:



The vertices are popped off the stack in the following order:

$e f g b c a d$.

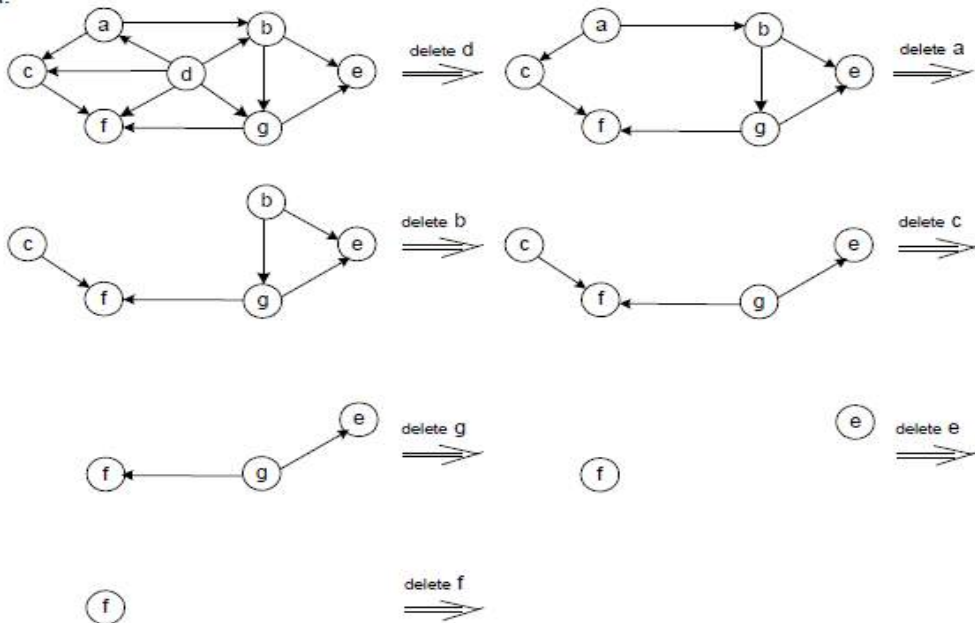
The topological sorting order obtained by reversing the list above is

$d a c b g f e$.

13. [교재 168쪽 Exercise 4.2 5번 변형] (5점)

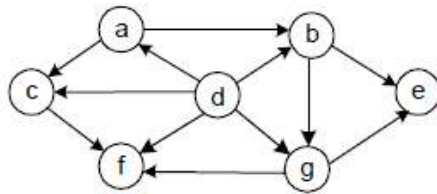
교재 168쪽 Exercise 4.2 1번의 그림 (a)에 source removal algorithm을 적용하여 topological sorting problem을 푸시오. (위 문제 12번과 같은 그림임)

5. a.



The topological ordering obtained is $d a b c g e f$.

1. a. The digraph and the stack of its DFS traversal that starts at vertex a are given below:



$$\begin{array}{ccccc} & & f & & \\ & e & g & & \\ & b & & c & \\ a & & & & d \end{array}$$

The vertices are popped off the stack in the following order:

$e\ f\ g\ b\ c\ a\ d.$

The topological sorting order obtained by reversing the list above is

$d\ a\ c\ b\ g\ f\ e.$

14. [교재 174쪽 Exercise 4.3 2-(a),(b)번] (총 10점, 부분문제 (a~b) 각 5점)
 부분문제 (a) ~ (b)를 푸시오. [부분문제 c는 풀 필요 없음.]
2. a. The permutations of $\{1, 2, 3, 4\}$ generated by the bottom-up minimal-change algorithm:

start	1
insert 2 into 1 right to left	12 21
insert 3 into 12 right to left	123 132 312
insert 3 into 21 left to right	321 231 213
insert 4 into 123 right to left	1234 1243 1423 4123
insert 4 into 132 left to right	4132 1432 1342 1324
insert 4 into 312 right to left	3124 3142 3412 4312
insert 4 into 321 left to right	4321 3421 3241 3214
insert 4 into 231 right to left	2314 2341 2431 4231
insert 4 into 213 left to right	4213 2413 2143 2134

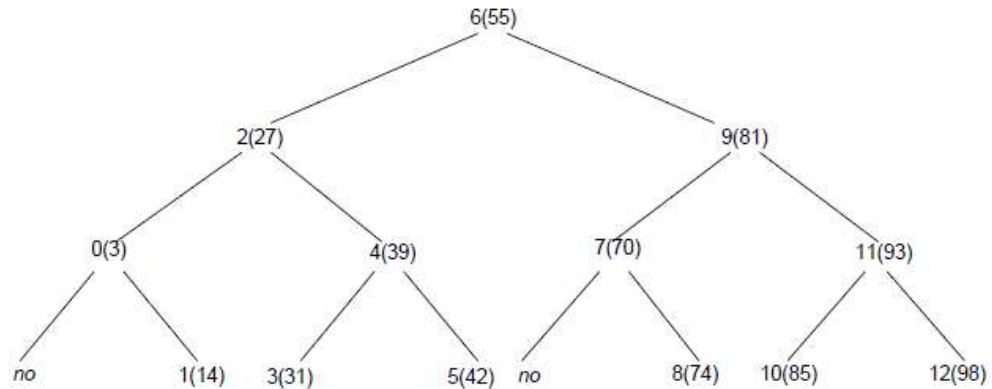
- b. The permutations of $\{1, 2, 3, 4\}$ generated by the Johnson-Trotter algorithm. (Read horizontally; the largest mobile element is shown in bold.)

<u>1</u> <u>2</u> <u>3</u> <u>4</u>	<u>1</u> <u>2</u> <u>4</u> <u>3</u>	<u>1</u> <u>4</u> <u>2</u> <u>3</u>	<u>4</u> <u>1</u> <u>2</u> <u>3</u>
<u>4</u> <u>1</u> <u>3</u> <u>2</u>	<u>1</u> <u>4</u> <u>3</u> <u>2</u>	<u>1</u> <u>3</u> <u>4</u> <u>2</u>	<u>1</u> <u>3</u> <u>2</u> <u>4</u>
<u>3</u> <u>1</u> <u>2</u> <u>4</u>	<u>3</u> <u>1</u> <u>4</u> <u>2</u>	<u>3</u> <u>4</u> <u>1</u> <u>2</u>	<u>4</u> <u>3</u> <u>1</u> <u>2</u>
<u>4</u> <u>3</u> <u>2</u> <u>1</u>	<u>3</u> <u>4</u> <u>2</u> <u>1</u>	<u>3</u> <u>2</u> <u>4</u> <u>1</u>	<u>3</u> <u>2</u> <u>1</u> <u>4</u>
<u>2</u> <u>3</u> <u>1</u> <u>4</u>	<u>2</u> <u>3</u> <u>4</u> <u>1</u>	<u>2</u> <u>4</u> <u>3</u> <u>1</u>	<u>4</u> <u>2</u> <u>3</u> <u>1</u>
<u>4</u> <u>2</u> <u>1</u> <u>3</u>	<u>2</u> <u>4</u> <u>1</u> <u>3</u>	<u>2</u> <u>1</u> <u>4</u> <u>3</u>	<u>2</u> <u>1</u> <u>3</u> <u>4</u>

15. [교재 182쪽 Exercise 4.4 3번] (총 20점, 부분문제 (a~d) 각 5점)
 부분문제 (a) ~ (d)를 푸시오. [키 비교횟수는 3-way comparison을 1회로 세도록 한다.]

3. a. According to formula (4.5), $C_{worst}(13) = \lceil \log_2(13 + 1) \rceil = 4$.

b. In the comparison tree below, the first number indicates the element's index, the second one is its value:



The searches for each of the elements on the last level of the tree, i.e., the elements in positions 1(14), 3(31), 5(42), 8(74), 10(85), and 12(98) will require the largest number of key comparisons.

c. $C_{avg}^{yes} = \frac{1}{13} \cdot 1 \cdot 1 + \frac{1}{13} \cdot 2 \cdot 2 + \frac{1}{13} \cdot 3 \cdot 4 + \frac{1}{13} \cdot 4 \cdot 6 = \frac{41}{13} \approx 3.2$.

d. $C_{avg}^{no} = \frac{1}{14} \cdot 3 \cdot 2 + \frac{1}{14} \cdot 4 \cdot 12 = \frac{54}{14} \approx 3.9$.

16. [교재 192쪽 Exercise 4.5 2번] (10점)

<Sol>

$n=7$ 이므로, median은 $k = \left\lceil \frac{7}{2} \right\rceil = 4$ 번째 원소임. $k-1 = 3$.

Lomuto partition을 이용한 quickselect을 8, 15, 3, 19, 24, 32, 6 리스트에 적용하면, 아래와 같이 partition이 일어난다. ($l=0$, $r=6$, $k=4$)

	[0]	[1]	[2]	[3] = k-1	[4]	[5]	[6]
초기 순서	8	15	3	19	24	32	6
첫 번째 파티션	s	i					
	8	3	15	19	24	32	6
		s	i				
	8	3	15	19	24	32	6
		s					i
	8	3	15	19	24	32	6
		s					i
	8	3	6	19	24	32	15
		s					
	6	3	8	19	24	32	15

첫 번째 파티션을 한 결과, $s = 2 < l+k-1 (=3)$ 이므로, $s+1$ 부터 오른쪽 끝까지 자리의 숫자들의 리스트로 두 번째 파티션을 수행함. ($l=3$, $r=6$, $k=1$)

	[0]	[1]	[2]	[3]	[4]	[5]	[6]
초기 순서	8	15	3	19	24	32	6
첫 번째 파티션 후	6	3	8	19	24	32	15
두 번째 파티션	6	3	8	s	i		
				19	24	32	15
	6	3	8	s			i
				19	24	32	15
	6	3	8		s		i
				19	15	32	24
	6	3	8		s		
				15	19	32	24

두 번째 파티션을 한 결과, $s = 4 > l+k-1 (=3)$ 이므로, 리스트의 왼쪽 부분을 가지고 계속 진행하나, ($l=3, r=3, k=1$) 그 리스트가 단 하나의 원소 15만을 가지므로, 이것이 median이 된다.

	[0]	[1]	[2]	[3]	[4]	[5]	[6]
두 번째 파티션 후	6	3	8	15	19	32	24
세 번째 파티션	6	3	8	s			
				15	19	32	24
				↑ median			