

12장 연습문제 정답

01.

풀이 생략(다양한 답이 있을 수 있다.)

02.

(a)

```
algorithm max(int a, int b){  
    if a >= b then  
        print a;  
    else  
        print b;  
    endif  
}
```

(b)

```
algorithm func(int a, int b){  
    if a > b then  
        r = a + b;  
    else if a < b then  
        r = b - a;  
    else  
        r = 0;  
    endif  
  
    return r;  
}
```

(c)

```
algorithm swap(int a, int b){  
    int temp;  
  
    temp = a;  
    a = b;  
    b = temp;  
}
```

(d)

```
algorithm sum(int a, int b){
    int r = 0;

    if a <= b then
        for i = a to b
            r += i;
        next i
    else
        for i = b to a
            r += i;
        next i
    endif
}
```

(e)

```
algorithm odd_even(int n){
    int odd[];
    int even[];
    int i = 0;
    int j = 0;
    for k = 1 to n
        if k % 2 == 0 then
            even[i] = k;
            i++;
        else
            odd[j] = k;
            j++;
        endif
    next k
}
```

(f)

```
algorithm matrix(int A[], int B[]){
    int PLUS[][];
    int MINUS[][];

    for i = 0 to a-1
        for j = 0 to b-1
            PLUS[i][j] = A[i][j] + B[i][j];
            MINUS[i][j] = A[i][j] - B[i][j];
        next j
    next i
}
```

(g)

```
algorithm matrix(int A[], int B[]){
    int RESULT[];

    for i = 0 to n-1
        for j = 0 to n-1
            RESULT[i][j] = 0;
            for k = 0 to n-1
                RESULT[i][j] += A[i][k] * B[k][j];
            next k
        next j
    next i
}
```

03.

(a) $n \geq 1$ 일 때 $\log n \leq n$, $3\log n \leq 3n$

$n \geq 1$ 인 모든 자연수에 대해 성립, $c = 3$ 로 복잡도는 $O(n)$
 \therefore 성립한다.

$$(b) f(n) = \frac{n^4 + 2n^2 + 1}{n^2 + 1} = n^2 + 1, \quad n^2 + 1 \leq n^2 + n^2 = 2n^2$$

$n \geq 1$ 인 모든 자연수에 대해 성립, $c = 2$ 로 복잡도는 $O(n^2)$
 \therefore 성립한다.

$$(c) f(n) = \frac{x^3 + 2}{x + 1} = \frac{(x+1)(x^2 - x + 1) + 1}{x + 1} = x^2 - x + 1 + \frac{1}{x + 1} \leq x^2$$

$n \geq 2$ 인 모든 자연수에 대해 성립, $c = 1$ 로 복잡도는 $O(x^2)$
 \therefore 성립하지 않는다.

$$(d) \sqrt{n} \leq n$$

$n \geq 1$ 인 모든 자연수에 대해 성립, $c = 1$ 로 복잡도는 $O(n)$
 \therefore 성립하지 않는다.

$$(e) \log n < n \text{ 이므로 } f(n) = \log n + 2n^2 < n + 2n^2 \leq n^2 + 2n^2 = 3n^2$$

$n \geq 1$ 인 모든 자연수에 대해 성립, $c = 3$ 로 복잡도는 $O(n^2)$
 \therefore 성립하지 않는다.

04.

- (a) 성립한다.
- (b) 성립하지 않는다.
- (c) 성립한다.
- (d) 성립한다.
- (e) 성립한다.
- (f) 성립하지 않는다.

05.

(a) $f(n) = n + 3 \leq n + 3n = 4n$

$\therefore n \geq 1$ 인 모든 자연수에 대해 성립, $c = 4$, $k = 1$, $O(n)$

(b) $f(n) = \frac{3}{2n} \leq n$

$\therefore n \geq 2$ 인 모든 자연수에 대해 성립, $c = 1$, $k = 2$, $O(n)$

(c) $f(n) = 3 + \frac{1}{2n} \leq 4n$

$\therefore n \geq 1$ 인 모든 자연수에 대해 성립, $c = 4$, $k = 1$, $O(n)$

(d) $f(n) = n^2 + 5n + \frac{1}{2} \leq n^2 + 5n^2 + \frac{1}{2}n^2 = 2n^2 + 10n^2 + n^2 = 13n^2$

$\therefore n \geq 1$ 인 모든 자연수에 대해 성립, $c = 13$, $k = 1$, $O(n^2)$

(e) $n \geq 3$ 일 때 $7n \leq n^3$ 므로 $f(n) = n^3 + 7n \leq n^3 + n^3 = 2n^3$

$\therefore n \geq 3$ 인 모든 자연수에 대해 성립, $c = 2$, $k = 3$, $O(n^3)$

(f) $\log n < n$ 이고 $n \log n < n^2$ 이므로 $f(n) = \log n + n \log n < n + n^2 \leq n^2 + n^2 = 2n^2$

$\therefore n \geq 1$ 인 모든 자연수에 대해 성립, $c = 2$, $k = 1$, $O(n^2)$

06.

(a) $n \geq 1$ 인 모든 자연수에 대해 성립, $c = 3$, $k = 1$, $O(n^2)$

(b) $n \geq 2$ 인 모든 자연수에 대해 성립, $c = 2$, $k = 2$, $O(n^2)$

(c) $n \geq 1$ 인 모든 자연수에 대해 성립, $c = 2$, $k = 1$, $O(n^3)$

(d) $n \geq 1$ 인 모든 자연수에 대해 성립, $c = 1$, $k = 1$, $O(n)$

(e) $n \geq 4$ 인 모든 자연수에 대해 성립, $c = 2$, $k = 4$, $O(2^n)$

(f) $n \geq 1$ 인 모든 자연수에 대해 성립, $c = 2$, $k = 1$, $O(2^{3n})$

07.

- (a) 실행문은 `return k+1;` 하나이므로 전체 연산횟수는 1 $\therefore f(n) = 1 \quad \therefore O(1)$
- (b) 실행문은 `input n;`, `x = k + n;`, `print x;` 세 개이며 `x = k + n;`는 +연산과 대입 연산 두 개이므로 전체 연산횟수는 4개 $\therefore f(n) = n \quad \therefore O(n)$
- (c) 실행문은 `x += n;`, `return x;` 두 개인데 `x += n;`의 경우 $n > 10$ 일 경우에만 실행되며 +연산과 대입 연산 두 개가 실행된다. 그러므로 전체 실행 횟수는 최대 2회 $\therefore f(n) = n \quad \therefore O(n)$
- (d) 실행문은 `input j;`, `x += j;`, `return x;` 세 개인데, `x += j;`는 $j > n$ 일 때 n 번 실행되며 +연산과 대입 연산 두 개가 실행된다. 그러므로 전체 실행 횟수는 최대 $n+4$ 회 $\therefore f(n) = n \quad \therefore O(n)$
- (e) `return`문 1회 실행되므로 \therefore 복잡도는 $O(1)$

08.

- (a) $n \geq 1$ 인 모든 자연수에 대해 성립, $c = 4$, 복잡도는 $O(n^2)$
- (b) $n \geq 1$ 인 모든 자연수에 대해 성립, $c = 8$, 복잡도는 $O(n^2)$
- (c) $n \geq 1$ 인 모든 자연수에 대해 성립, $c = 4$, 복잡도는 $O(n^2)$
- (d) $n \geq 1$ 인 모든 자연수에 대해 성립, $c = 4$, 복잡도는 $O(n^2)$
- (e) $n \geq 1$ 인 모든 자연수에 대해 성립, $c = 5$, 복잡도는 $O(n^2)$

09.

- (a) $\gcd(162, 102) = \gcd(102, 60) = \gcd(60, 42) = \gcd(42, 18) = \gcd(18, 6) = \gcd(6, 0)$
 \therefore 최대공약수 : 6
- (b) $\gcd(255, 165) = \gcd(165, 90) = \gcd(90, 75) = \gcd(75, 15) = \gcd(15, 0)$
 \therefore 최대공약수 : 15
- (c) $\gcd(402, 388) = \gcd(388, 14) = \gcd(14, 10) = \gcd(10, 4) = \gcd(4, 2) = \gcd(2, 0)$
 \therefore 최대공약수 : 2
- (d) $\gcd(469, 123) = \gcd(123, 100) = \gcd(100, 23) = \gcd(23, 8) = \gcd(8, 7)$
 $= \gcd(7, 1) = \gcd(1, 0) \quad \therefore$ 최대공약수 : 1
- (e) 최대공약수 : 165
- (f) 최대공약수 : 432
- (g) 최대공약수 : 123
- (h) 최대공약수 : 1

10.

$$(a) \begin{cases} f(1) = 2 \\ f(n) = 4 + f(n-1) \quad n > 1 \end{cases}$$

n부터 1이 될 때까지 함수 f(n)를 호출하며 4를 더한다.

$$f(n) = n \quad \therefore O(n)$$

$$(b) \begin{cases} f(1) = 3 \\ f(n) = 3 \times f(n-1) \quad n > 1 \end{cases}$$

n부터 1이 될 때까지 함수 f(n)를 호출하며 3을 곱한다.

$$f(n) = n \quad \therefore O(n)$$

$$(c) f(n) = n \quad \therefore O(n)$$

$$(d) f(n) = 2n + 1 \quad \therefore O(n)$$

11.

(a)

```
algorithm f(int n){
    if n ≥ 1 then
        return n * f(n-1);
    else
        return 1;
    endif
}
```

n부터 1까지 함수를 호출하여 *연산을 수행하고
0일 때는 1을 리턴한다.

$$\therefore f(n) = n + 1 \quad O(n)$$

(b)

```
algorithm f(int n){
    if n > 1 then
        return f(n-1);
    else
        return 0;
    endif
}
```

n부터 2까지 함수를 호출하고 1일 때는 0을 리턴한다.

$$\therefore f(n) = n \quad O(n)$$

(c)

```
algorithm f(int n){
    if n > 2 then
        return n + f(n-2);
    else if n = 2 then
        return 1;
    else if n = 1 then
        return 0;
    endif
}
```

$$f(n) = n + 1 \quad O(n)$$

(d)

```
algorithm f(int n){
    if n > 2 then
        return f(n-2);
    else
        return 1;
    endif
}
```

$$f(n) = n - 1 \quad O(n)$$

(e)

```
algorithm f(int n){
    if n > 2 then
        return f(n-1) * f(n-2);
    else
        return 1;
    }
}
```

$$f(n) = 2n \quad O(n)$$

12.

```
algorithm gcd(int a, int b){
    if b == 0 then
        return a;
    else
        gcd(b, a mod b);
    endif
}
```

13.

```
algorithm binary_search(list, target, int s, int e){
    m = (s + e) / 2;
    if list[m] == target then
        return m;
    else if list[m] > target then
        return binary_search(list, target, s, m-1);
    else if list[m] < target then
        return binary_search(list, target, m+1, e);
    else
        return 0;
    endif
}
```

14.

(a)

- ① (1) 인덱스 0번과 B를 비교, $C \neq B$
(2) 인덱스 1번과 B를 비교, $K \neq B$
(3) 인덱스 2번과 B를 비교, $O \neq B$
(4) 인덱스 3번과 B를 비교, $A \neq B$
(5) 인덱스 4번과 B를 비교, $M \neq B$
(6) 인덱스 5번과 B를 비교, $R \neq B$
(7) 인덱스 6번과 B를 비교, $B = B$
 \therefore B의 인덱스는 6

- ② (1) 인덱스 0번과 B를 비교, $J \neq B$
(2) 인덱스 1번과 B를 비교, $I \neq B$
(3) 인덱스 2번과 B를 비교, $H \neq B$
(4) 인덱스 3번과 B를 비교, $G \neq B$
(5) 인덱스 4번과 B를 비교, $F \neq B$
(6) 인덱스 5번과 B를 비교, $E \neq B$
(7) 인덱스 6번과 B를 비교, $D \neq B$
(8) 인덱스 7번과 B를 비교, $C \neq B$
(9) 인덱스 8번과 B를 비교, $B \neq B$
 \therefore B의 인덱스는 8

③ B의 인덱스는 4

④ B의 인덱스는 2

⑤ B의 인덱스는 1

(b)

① 정렬된 집합이 아니므로 이진탐색을 사용할 수 없다.

② (1) 인덱스 0부터 9까지 10개의 원소들이 있으므로 $\left\lfloor \frac{0+10}{2} \right\rfloor = 5$

(2) 인덱스 5의 E는 B보다 크므로 남아 있는 다른 원소들과 탐색 $\left\lfloor \frac{6+10}{2} \right\rfloor = 8$

(3) 인덱스 8의 B는 B와 같으므로 탐색 종료

\therefore B의 인덱스는 8

③ 이진탐색을 사용할 수 없다.

④ 이진탐색을 사용할 수 없다.

⑤ B의 인덱스는 1

15.

(a)

- ① - 인덱스 0의 11과 인덱스 1의 7 비교, $11 > 7$ 이므로 교체
 { 7, 11, 3, 15, 9, 2, 20, 4, 18, 10 }
- 인덱스 1의 1과 인덱스 2의 3 비교, $7 > 3$ 이므로 교체
 { 7, 3, 11, 15, 9, 2, 20, 4, 18, 10 }
- 인덱스 2의 11과 인덱스 3의 15 비교, $11 < 15$ 이므로 유지
 { 7, 3, 11, 15, 9, 2, 20, 4, 18, 10 }
- 인덱스 3의 15와 인덱스 4의 9 비교, $15 > 9$ 이므로 교체
 { 7, 3, 11, 9, 15, 2, 20, 4, 18, 10 }
- 인덱스 4의 15와 인덱스 5의 2 비교, $15 > 2$ 이므로 교체
 { 7, 3, 11, 9, 2, 15, 20, 4, 18, 10 }
- 인덱스 5의 15와 인덱스 6의 20 비교 $15 < 20$ 이므로 유지
 { 7, 3, 11, 9, 2, 15, 20, 4, 18, 10 }
- 인덱스 6의 20과 인덱스 7의 4 비교 $20 > 4$ 이므로 교체
 { 7, 3, 11, 9, 2, 15, 4, 20, 18, 10 }
- 인덱스 7의 20과 인덱스 8의 18 비교 $20 > 18$ 이므로 교체
 { 7, 3, 11, 9, 2, 15, 4, 18, 20, 10 }
- 인덱스 8의 20과 인덱스 9의 10 비교 $20 > 10$ 이므로 교체
 { 7, 3, 11, 9, 2, 15, 4, 18, 10, 20 }
- 인덱스 0의 7과 인덱스 1의 3 비교, $7 > 3$ 이므로 교체
 { 3, 7, 11, 9, 2, 15, 4, 18, 10, 20 }
- 인덱스 1의 7과 인덱스 2의 11 비교, $7 < 11$ 이므로 유지
 { 3, 7, 11, 9, 2, 15, 4, 18, 10, 20 }
- 인덱스 2의 11과 인덱스 3의 9 비교, $11 > 9$ 이므로 교체
 { 3, 7, 9, 11, 2, 15, 4, 18, 10, 20 }
- 인덱스 3의 11과 인덱스 4의 2 비교, $11 > 2$ 이므로 교체
 { 3, 7, 9, 2, 11, 15, 4, 18, 10, 20 }
- 인덱스 4의 11과 인덱스 5의 15 비교, $11 < 15$ 이므로 유지
 { 3, 7, 9, 2, 11, 15, 4, 18, 10, 20 }
- 인덱스 5의 15와 인덱스 6의 4 비교, $15 > 4$ 이므로 교체
 { 3, 7, 9, 2, 11, 4, 15, 18, 10, 20 }
- 인덱스 6의 15와 인덱스 7의 18 비교, $15 < 18$ 이므로 유지
 { 3, 7, 9, 2, 11, 4, 15, 18, 10, 20 }
- 인덱스 7의 18과 인덱스 8의 10 비교, $18 > 10$ 이므로 교체
 { 3, 7, 9, 2, 11, 4, 15, 10, 18, 20 }
- 인덱스 8의 18과 인덱스 9의 20 비교, $18 < 20$ 이므로 유지
 { 3, 7, 9, 2, 11, 4, 15, 10, 18, 20 }
- 인덱스 0의 3과 인덱스 1의 7 비교, $3 < 7$ 이므로 유지
 { 3, 7, 9, 2, 11, 4, 15, 10, 18, 20 }
- 인덱스 1의 7과 인덱스 2의 9 비교, $7 < 9$ 이므로 유지
 { 3, 7, 9, 2, 11, 4, 15, 10, 18, 20 }
- 인덱스 2의 9와 인덱스 3의 2 비교, $9 > 2$ 이므로 교체
 { 3, 7, 2, 9, 11, 4, 15, 10, 18, 20 }

- 인덱스 3의 9와 인덱스 4의 11 비교, $9 < 11$ 이므로 유지
 $\{ 3, 7, 2, 9, 11, 4, 15, 10, 18, 20 \}$
- 인덱스 4의 11과 인덱스 5의 4 비교, $11 > 4$ 이므로 교체
 $\{ 3, 7, 2, 9, 4, 11, 15, 10, 18, 20 \}$
- 인덱스 5의 11과 인덱스 6의 15 비교, $11 < 15$ 이므로 유지
 $\{ 3, 7, 2, 9, 4, 11, 15, 10, 18, 20 \}$
- 인덱스 6의 15와 인덱스 7의 10 비교, $15 > 10$ 이므로 교체
 $\{ 3, 7, 2, 9, 4, 11, 10, 15, 18, 20 \}$
- 인덱스 7의 15와 인덱스 8의 18 비교, $15 < 18$ 이므로 유지
 $\{ 3, 7, 2, 9, 4, 11, 10, 15, 18, 20 \}$
- 인덱스 8의 18과 인덱스 9의 20 비교, $18 < 20$ 이므로 유지
 $\{ 3, 7, 2, 9, 4, 11, 10, 15, 18, 20 \}$
- 인덱스 0의 3과 인덱스 1의 7 비교, $0 < 7$ 이므로 유지
 $\{ 3, 7, 2, 9, 4, 11, 10, 15, 18, 20 \}$
- 인덱스 1의 7과 인덱스 2의 2 비교, $7 > 2$ 이므로 교체
 $\{ 3, 2, 7, 9, 4, 11, 10, 15, 18, 20 \}$
- 인덱스 2의 7과 인덱스 3의 9 비교, $7 < 9$ 이므로 유지
 $\{ 3, 2, 7, 9, 4, 11, 10, 15, 18, 20 \}$
- 인덱스 3의 9와 인덱스 4의 4 비교, $9 > 4$ 이므로 교체
 $\{ 3, 2, 7, 4, 9, 11, 10, 15, 18, 20 \}$
- 인덱스 4의 9와 인덱스 5의 11 비교, $9 < 11$ 이므로 유지
 $\{ 3, 2, 7, 4, 9, 11, 10, 15, 18, 20 \}$
- 인덱스 5의 11과 인덱스 6의 10 비교, $11 > 10$ 이므로 교체
 $\{ 3, 2, 7, 4, 9, 10, 11, 15, 18, 20 \}$
- 인덱스 6의 11과 인덱스 7의 15 비교, $11 < 15$ 이므로 유지
 $\{ 3, 2, 7, 4, 9, 10, 11, 15, 18, 20 \}$
- 인덱스 7의 15와 인덱스 8의 18 비교, $15 < 18$ 이므로 유지
 $\{ 3, 2, 7, 4, 9, 10, 11, 15, 18, 20 \}$
- 인덱스 8의 18와 인덱스 9의 20 비교, $18 < 20$ 이므로 유지
 $\{ 3, 2, 7, 4, 9, 10, 11, 15, 18, 20 \}$
- 인덱스 0의 3과 인덱스 1의 2 비교, $3 > 2$ 이므로 교체
 $\{ 2, 3, 7, 4, 9, 10, 11, 15, 18, 20 \}$
- 인덱스 1의 3과 인덱스 2의 7 비교, $3 < 7$ 이므로 유지
 $\{ 2, 3, 7, 4, 9, 10, 11, 15, 18, 20 \}$
- 인덱스 2의 7과 인덱스 3의 4 비교, $7 > 4$ 이므로 교체
 $\{ 2, 3, 4, 7, 9, 10, 11, 15, 18, 20 \}$
- 이후로 처음부터 마지막 인덱스까지 비교 후 정렬 종료

② $\{ N, L, O, A, G, E, V, P, J, M \}$

- $N > L : \{ L, N, O, A, G, E, V, P, J, M \}$
- $N < O : \{ L, N, O, A, G, E, V, P, J, M \}$
- $O > A : \{ L, N, A, O, G, E, V, P, J, M \}$
- $O > G : \{ L, N, A, G, O, E, V, P, J, M \}$
- $O > E : \{ L, N, A, G, E, O, V, P, J, M \}$
- $O < V : \{ L, N, A, G, E, O, V, P, J, M \}$

- $V > P : \{ L, N, A, G, E, O, P, V, J, M \}$
- $V > J : \{ L, N, A, G, E, O, P, J, V, M \}$
- $V > M : \{ L, N, A, G, E, O, P, J, M, V \}$
- $L < N : \{ L, N, A, G, E, O, P, J, M, V \}$
- $N > A : \{ L, A, N, G, E, O, P, J, M, V \}$
- $N > G : \{ L, A, G, N, E, O, P, J, M, V \}$
- $N > E : \{ L, A, G, E, N, O, P, J, M, V \}$
- $N < O : \{ L, A, G, E, N, O, P, J, M, V \}$
- $O < P : \{ L, A, G, E, N, O, P, J, M, V \}$
- $P > J : \{ L, A, G, E, N, O, J, P, M, V \}$
- $P > M : \{ L, A, G, E, N, O, J, M, P, V \}$
- $P < V : \{ L, A, G, E, N, O, J, M, P, V \}$
- $L > A : \{ A, L, G, E, N, O, J, M, P, V \}$
- $L > G : \{ A, G, L, E, N, O, J, M, P, V \}$
- $L > E : \{ A, G, E, L, N, O, J, M, P, V \}$
- $L < N : \{ A, G, E, L, N, O, J, M, P, V \}$
- $N < O : \{ A, G, E, L, N, O, J, M, P, V \}$
- $O > J : \{ A, G, E, L, N, J, O, M, P, V \}$
- $O > M : \{ A, G, E, L, N, J, M, O, P, V \}$
- $O < P : \{ A, G, E, L, N, J, M, O, P, V \}$
- $P < V : \{ A, G, E, L, N, J, M, O, P, V \}$
- $A < G : \{ A, G, E, L, N, J, M, O, P, V \}$
- $G > E : \{ A, E, G, L, N, J, M, O, P, V \}$
- $G < L : \{ A, E, G, L, N, J, M, O, P, V \}$
- $L < N : \{ A, E, G, L, N, J, M, O, P, V \}$
- $N > J : \{ A, E, G, L, J, N, M, O, P, V \}$
- $N > M : \{ A, E, G, L, J, M, N, O, P, V \}$
- $N < O : \{ A, E, G, L, J, M, N, O, P, V \}$
- $O < P : \{ A, E, G, L, J, M, N, O, P, V \}$
- $P < V : \{ A, E, G, L, J, M, N, O, P, V \}$
- $A < E : \{ A, E, G, L, J, M, N, O, P, V \}$
- $E < G : \{ A, E, G, L, J, M, N, O, P, V \}$
- $G < L : \{ A, E, G, L, J, M, N, O, P, V \}$
- $L > J : \{ A, E, G, J, L, M, N, O, P, V \}$
- $L < M : \{ A, E, G, J, L, M, N, O, P, V \}$
- $M < N : \{ A, E, G, J, L, M, N, O, P, V \}$
- $N < O : \{ A, E, G, J, L, M, N, O, P, V \}$
- $O < P : \{ A, E, G, J, L, M, N, O, P, V \}$
- $P < V : \{ A, E, G, J, L, M, N, O, P, V \}$
- 이후로 처음부터 마지막 인덱스까지 비교 후 정렬 종료

③ $\{ A, B, E, H, I, J, L, M, P, X \}$

- 이후로 처음부터 마지막 인덱스까지 비교 후 정렬 종료

④ { 10, 19, 23, 26, 32, 44, 60, 78 }

- 이후로 처음부터 마지막 인덱스까지 비교 후 정렬 종료

⑤ {algorithm, binary, bubble, euclidean, index, insertion, merge, quick, recursion}

(b)

① 초기집합 : { 11, 7, 3, 15, 9, 2, 20, 4, 18, 10 }

- { 11 }

- $7 < 11$: { 7, 11 }

- $3 < 7 < 11$: { 3, 7, 11 }

- $11 < 15$: { 3, 7, 11, 15 }

- $7 < 9 < 11$: { 3, 7, 9, 11, 15 }

- $2 < 3$: { 2, 3, 7, 9, 11, 15 }

- $15 < 20$: { 2, 3, 7, 9, 11, 15, 20 }

- $3 < 4 < 7$: { 2, 3, 4, 7, 9, 11, 15, 20 }

- $15 < 18 < 20$: { 2, 3, 7, 9, 11, 15, 18, 20 }

- $9 < 10 < 11$: { 2, 3, 7, 9, 10, 11, 15, 18, 20 }

② 초기 : { N, L, O, A, G, E, V, P, J, M }

- { N }

- { L, N }

- { L, N, O }

- { A, L, N, O }

- { A, G, L, N, O }

- { A, E, G, L, N, O }

- { A, E, G, L, N, O, V }

- { A, E, G, L, N, O, P, V }

- { A, E, G, J, L, N, O, P, V }

- { A, E, G, J, L, M, N, O, P, V }

③ { A, B, E, H, I, J, L, M, P, X }

④ { 10, 19, 23, 26, 32, 44, 60, 78 }

⑤ {algorithm, binary, bubble, euclidean, index, insertion, merge, quick, recursion}

(c)

①

- 기준값 : 11(인덱스 0)

11보다 큰 값 : 15(인덱스 3) / 11보다 작은 값 : 10(인덱스 9) / 큰 값 인덱스 3 < 작은 값 인덱스 9

\therefore {11, 7, 3, 10, 9, 2, 20, 4, 18, 15}

- 기준값 : 11(인덱스 0)

11보다 큰 값 : 20(인덱스 6) / 11보다 작은 값 : 4(인덱스 7) / 큰 값 인덱스 6 < 작은 값 인덱스 7

\therefore {11, 7, 3, 10, 9, 2, 4, 20, 18, 15}

- 기준값 : 11(인덱스 0)
11보다 큰 값 : 20(인덱스 7) / 11보다 작은 값 : 4(인덱스 6) / 큰 값 인덱스 7 > 작은 값 인덱스 6
∴ {4, 7, 3, 10, 9, 2, 11, 20, 18, 15}
- 기준값 : 4(인덱스 0)
4보다 큰 값 : 7(인덱스 1) / 4보다 작은 값 : 3(인덱스 2) / 큰 값 인덱스 1 < 작은 값 인덱스 2
∴ {4, 3, 7, 10, 9, 2, 11, 20, 18, 15}
- 기준값 : 4(인덱스 0)
4보다 큰 값 : 7(인덱스 2) / 4보다 작은 값 : 3(인덱스 1) / 큰 값 인덱스 2 > 작은 값 인덱스 1
∴ {3, 4, 7, 10, 9, 2, 11, 20, 18, 15}
- 기준값 : 3(인덱스 0)
3보다 큰 값 : 4(인덱스 1) / 3보다 작은 값 : 2(인덱스 5) / 큰 값 인덱스 1 < 작은 값 인덱스 5
∴ {3, 2, 7, 10, 9, 4, 11, 20, 18, 15}
- 기준값 : 3(인덱스 0)
3보다 큰 값 : 7(인덱스 2) / 3보다 작은 값 : 2(인덱스 1) / 큰 값 인덱스 2 > 작은 값 인덱스 1
∴ {2, 3, 7, 10, 9, 4, 11, 20, 18, 15}
- 기준값 : 2(인덱스 0)
2보다 큰 값 : 3(인덱스 1) / 2보다 작은 값 : 없음
∴ 기준값을 인덱스 1로 변경
- 기준값 : 3(인덱스 1)
3보다 큰 값 : 7(인덱스 2) / 3보다 작은 값 : 없음
∴ 기준값을 인덱스 2로 변경
- 기준값 : 7(인덱스 2)
7보다 큰 값 : 10(인덱스 3) / 7보다 작은 값 : 4(인덱스 5) / 큰 값 인덱스 3 < 작은 값 인덱스 5
∴ {2, 3, 7, 4, 9, 10, 11, 20, 18, 15}
- 기준값 : 7(인덱스 2)
7보다 큰 값 : 9(인덱스 4) / 7보다 작은 값 : 4(인덱스 3) / 큰 값 인덱스 4 > 작은 값 인덱스 3
∴ {2, 3, 4, 7, 9, 10, 11, 20, 18, 15}
- 기준값 : 4(인덱스 2)
4보다 큰 값 : 7(인덱스 3) / 4보다 작은 값 : 없음
∴ 기준값을 인덱스 3으로 변경
- 기준값 : 7(인덱스 3)
7보다 큰 값 : 9(인덱스 4) / 7보다 작은 값 : 없음
∴ 기준값 인덱스 4로 변경

- 기준값 : 9(인덱스 4)
9보다 큰 값 : 10(인덱스 5) / 9보다 작은 값 : 없음
∴ 기준값 인덱스 5로 변경
- 기준값 : 10(인덱스 5)
10보다 큰 값 : 11(인덱스 6) / 10보다 작은 값 : 없음
∴ 기준값 인덱스 6으로 변경
- 기준값 11(인덱스 6)
11보다 큰 값 : 20(인덱스 7) / 11보다 작은 값 : 없음
∴ 기준값 인덱스 7로 변경
- 기준값 20(인덱스 7)
20보다 큰 값 : 없음 / 20보다 작은 값 : 15(인덱스 8)
∴ {2, 3, 4, 7, 9, 10, 11, 15, 18, 20}

② {N, L, O, A, G, E, V, P, J, M}

- 기준값 : N(인덱스 0)
N보다 큰 값 : O(인덱스 2) / N보다 작은 값 : M(인덱스 9) / 큰 값 인덱스 2 < 작은 값 인덱스 9
∴ {N, L, M, A, G, E, V, P, J, O}
- 기준값 : N(인덱스 0)
N보다 큰 값 : V(인덱스 6) / N보다 작은 값 : J(인덱스 8) / 큰 값 인덱스 6 < 작은 값 인덱스 8
∴ {N, L, M, A, G, E, J, P, V, O}
- 기준값 : N(인덱스 0)
N보다 큰 값 : P(인덱스 7) / N보다 작은 값 : J(인덱스 6) / 큰 값 인덱스 7 > 작은 값 인덱스 6
∴ {J, L, M, A, G, E, N, P, V, O}
- 기준값 : J(인덱스 0)
J보다 큰 값 : L(인덱스 1) / J보다 작은 값 : E(인덱스 5) / 큰 값 인덱스 1 < 작은 값 인덱스 5
∴ {J, E, M, A, G, L, N, P, V, O}
- 기준값 : J(인덱스 0)
J보다 큰 값 : M(인덱스 2) / J보다 작은 값 : G(인덱스 4) / 큰 값 인덱스 2 < 작은 값 인덱스 4
∴ {J, E, G, A, M, L, N, P, V, O}
- 기준값 : J(인덱스 0)
J보다 큰 값 : M(인덱스 4) / J보다 작은 값 : A(인덱스 3) / 큰 값 인덱스 4 > 작은 값 인덱스 3
∴ {A, E, G, J, M, L, N, P, V, O}
- 기준값 : A(인덱스 0)
A보다 큰 값 : E(인덱스 1) / A보다 작은 값 : 없음
∴ 기준값 인덱스 1로 변경

- 기준값 : E(인덱스 1)
E보다 큰 값 : G(인덱스 2) / E보다 작은 값 : 없음
∴ 기준값 인덱스 2로 변경
- 기준값 : G(인덱스 2)
G보다 큰 값 : J(인덱스 3) / G보다 작은 값 : 없음
∴ 기준값 인덱스 3으로 변경
- 기준값 : J(인덱스 3)
J보다 큰 값 : M(인덱스 4) / J보다 작은 값 : 없음
∴ 기준값 인덱스 4로 변경
- 기준값 : M(인덱스 4)
M보다 큰 값 : N(인덱스 6) / M보다 작은 값 : L(인덱스 5) / 큰 값 인덱스 6 > 작은 값 인덱스 5
∴ {A, E, G, J, L, M, N, P, V, O}
- 기준값 : L(인덱스 4)
L보다 큰 값 : M(인덱스 5) / L보다 작은 값 : 없음
∴ 기준값 인덱스 5로 변경
- 기준값 : M(인덱스 5)
M보다 큰 값 : N(인덱스 6) / M보다 작은 값 : 없음
∴ 기준값 인덱스 6으로 변경
- 기준값 : N(인덱스 6)
N보다 큰 값 : P(인덱스 7) / N보다 작은 값 : 없음
∴ 기준값 인덱스 7로 변경
- 기준값 : P(인덱스 6)
P보다 큰 값 : V(인덱스 8) / P보다 작은 값 : O(인덱스 9) / 큰 값 인덱스 8 < 작은 값 인덱스 9
∴ {A, E, G, J, L, M, N, P, O, V}
- 기준값 : P(인덱스 6)
P보다 큰 값 : V(인덱스 9) / P보다 작은 값 : O(인덱스 8) / 큰 값 인덱스 9 > 작은 값 인덱스 8
∴ {A, E, G, J, L, M, N, O, P, V}

③ {A, B, E, H, I, J, L, M, P, X}

④ { 10, 19, 23, 26, 32, 44, 60, 78 }

⑤ {algorithm, binary, bubble, euclidean, index, insertion, merge, quick, recursion}

(d)

①

{ 7, 11, 3, 15, 9, 2, 20, 4, 18, 10 }
 { 7, 11, 3, 15, 9 } { 2, 20, 4, 18, 10 }
 { 7, 11 } { 3, 15, 9 } { 2, 20 } { 4, 18, 10 }

$\{ 7 \}$ $\{ 11 \}$ $\{ 3 \}$ $\{ 15, 9 \}$ $\{ 2 \}$ $\{ 20 \}$ $\{ 4 \}$ $\{ 18, 10 \}$
 $\{ 7 \}$ $\{ 11 \}$ $\{ 3 \}$ $\{ 15 \}$ $\{ 9 \}$ $\{ 2 \}$ $\{ 20 \}$ $\{ 4 \}$ $\{ 18 \}$ $\{ 10 \}$
 $\{ 7, 11 \}$ $\{ 3, 15 \}$ $\{ 2, 9 \}$ $\{ 4, 20 \}$ $\{ 10, 18 \}$
 $\{ 3, 7, 11, 15 \}$ $\{ 2, 4, 9, 20 \}$ $\{ 10, 18 \}$
 $\{ 2, 3, 4, 7, 9, 11, 15, 20 \}$ $\{ 10, 18 \}$
 $\{ 2, 3, 4, 7, 9, 10, 11, 15, 18, 20 \}$

②

$\{ N, L, O, A, G, E, V, P, J, M \}$
 $\{ N, L, O, A, G \}$ $\{ E, V, P, J, M \}$
 $\{ N, L \}$ $\{ O, A, G \}$ $\{ E, V \}$ $\{ P, J, M \}$
 $\{ N \}$ $\{ L \}$ $\{ O \}$ $\{ A, G \}$ $\{ E, V \}$ $\{ P \}$ $\{ J, M \}$
 $\{ N \}$ $\{ L \}$ $\{ O \}$ $\{ A \}$ $\{ G \}$ $\{ E \}$ $\{ V \}$ $\{ P \}$ $\{ J \}$ $\{ M \}$
 $\{ L, N \}$ $\{ A, O \}$ $\{ E, G \}$ $\{ P, V \}$ $\{ J, M \}$
 $\{ A, L, N, O \}$ $\{ E, G, P, V \}$ $\{ J, M \}$
 $\{ A, E, G, L, N, O, P, V \}$ $\{ J, M \}$
 $\{ A, E, G, J, L, N, M, O, P, V \}$

③ $\{ A, B, E, H, I, J, L, M, P, X \}$

④ $\{ 10, 19, 23, 26, 32, 44, 60, 78 \}$

⑤ {algorithm, binary, bubble, euclidean, index, insertion, merge, quick, recursion}