

2021년 1학기 알고리즘분석 중간시험 01반

시작됨: 4월 29일 오후 1:00

설명

- 한 문제 씩 보입니다.
- 제한시간은 50분입니다. 시간관리에 주의해 주세요.
- 오픈북으로 진행합니다.
- 문제순서: 2점 5문항, 10점 6 문항 순서로 배치되어 있습니다.
- ecampus 내에서는 괄호 [] 를 별도의 기능으로 인식하므로 괄호 대신 < > 로 배열정보를 표시합니다.
- 답변이 숫자로만 적으라고 할 경우 숫자로만 적어야 합니다. 3을 삼 으로 적으면 오답 처리합니다. 또한 '3번', '3개' 등으로 작성할 경우 오답 처리됩니다.

문제 1

2점

greedy 방법이 항상 최적해를 찾는 것은 아니다.

☐ 참

☐ 거짓

문제 2

2점

동적계획법은 top-down 방식이다.

☐ 참

☐ 거짓

문제 3

2점

복잡도 함수들이 $n-100$, $n \log n + n$ 일 때 이 함수들은 $\Omega(n)$ 에 속한다.

☐ 참

☐ 거짓

문제 4

2점

이분검색은 divide-and-conquer 방식의 알고리즘이다.

☐ 참

☐ 거짓

문제 5**2점**

같은 문제를 recursion이 없는 알고리즘으로 해결하는 것 보다 recursion 이 있는 알고리즘으로 해결하는 것이 프로그램 수행 시간적인 측면에서 우수하다.

☐ 참

☐ 거짓

문제 6



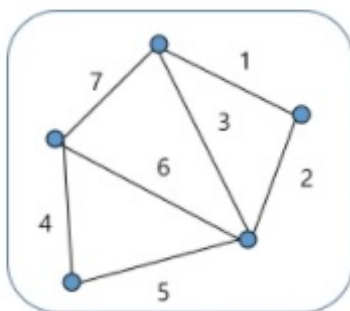
마감일에 가까울 때 퀴즈를 시작했으므로 퀴즈한 시간을 가지지 못합니다.

다음은 최소신장트리(MST)를 구하는 Kruskal 알고리즘의 의사코드이다.

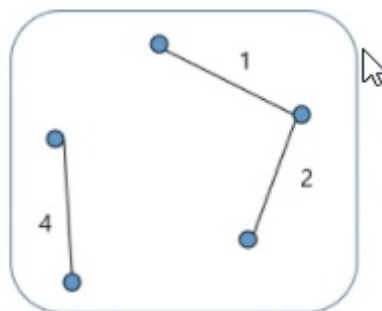
```
void kruskal(int n, int m, // 입력: 정점의 수 n, 에지의 수 m
            set_of_edges E, // 입력: 가중치를 포함한 이음선의 집합
            set_of_edges& F) { // 출력: MST를 이루는 이음선의 집합
    index i, j;
    set_pointer p, q;
    edge e;
    E에 속한 m개의 이음선을 가중치의 비내림차순으로 정렬;
    F =  $\phi$ ;
    initial(n);
    while (F에 속한 이음선의 개수가 n-1보다 작다) {
        e = 아직 점검하지 않은 최소의 가중치를 가진 이음선;
        (i, j) = e를 이루는 양쪽 정점의 인덱스;
        p = find(i);
        q = find(j);
        if (!equal(p, q)) {
            merge(p, q);
            e를 F에 추가;
        }
    }
}
```

다음 그림은 주어진 그래프와 이에 대해 Kruskal 알고리즘으로 MST를 구축 중인 모습이다.

에지 옆의 숫자는 에지의 길이를 표시한다.



(a) 주어진 그래프



(b) MST 구축 중인 F 표시

알고리즘의 시작부터 그림 (b)의 순간까지 함수 find()와 merge()의 호출횟수를 각각 적으시오. 숫자로 적을 것.

함수 find 호출 횟수 = 회

함수 merge 호출 횟수 = 회

문제 7

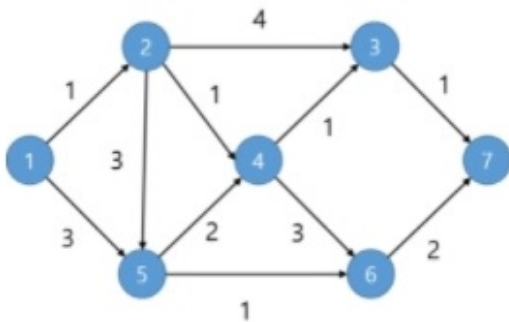
10점

다음은 플로이드 알고리즘의 의사코드와 D 행렬을 구하는 공식부분이다. D 는 교재의 정의를 따른다.

$$D^{(k)}[i][j] = \text{minimum} \{ D^{(k-1)}[i][j], D^{(k-1)}[i][k] + D^{(k-1)}[k][j] \}$$

```
void floyd2(int n, const number W[][], number D[][], index P[][]) {
    index i, j, k;
    for(i=1; i <= n; i++)
        for(j=1; j <= n; j++)
            P[i][j] = 0;
    D = W;
    for(k=1; k <= n; k++)
        for(i=1; i <= n; i++)
            for(j=1; j <= n; j++)
                if (D[i][k] + D[k][j] < D[i][j]) {
                    P[i][j] = k;
                    D[i][j] = D[i][k] + D[k][j];
                }
}
```

다음 그래프에 대해



$D^{(4)} < 1 > < 7 >$ 값은 무엇인가?

이 알고리즘이 종료된 후 $P < 1 > < 7 >$ 값은 무엇인가?

문제 8

10점

다음의 재현식에 대해

$$T(n) = T(n/2) + n/2, T(1)=1, n = 2^k, k \text{는 자연수}$$

$T(n)$ 을 n 의 함수로 표현한다.

$$T(n) = an^2 + bn + c$$

계수 a, b, c 의 값은 무엇인가? 계수는 0을 포함하여 정수로 나타낸다.

빈칸에 정수 숫자로 답할 때 만일 답이 0 인 경우에도 반드시 0으로 입력해야 한다.

음수인 경우 -와 숫자를 붙여서 기입할 것.

$a =$

$b =$

$c =$

문제 9

10점

8개의 데이터가 배열에 정렬되어 저장되어 있다. 배열 내에 없는 데이터를 이진검색 방법으로 찾을 때, 데이터가 배열 내에 없다는 것을 알기 위해 필요한 최소 데이터 비교 횟수, 최대 데이터 비교 횟수는 무엇인가? 숫자로 적으시오.

빈 공간을 확인하는 것은 데이터 비교 횟수에 포함하지 않는다. 배열 내의 중간 위치가 두 곳 될 경우 앞의 것을 중간 위치로 한다.

최소 = 회

최대 = 회

문제 10

10점

연쇄 행렬의 최소 곱셈 수 구하기 문제에서 5개의 행렬의 크기는 다음과 같다.

$A_1: 3 \times 4, A_2: 4 \times 2, A_3: 2 \times 3, A_4: 3 \times 3, A_5: 3 \times 1$

A_k 의 크기는 $d_{k-1} \times d_k$

$M[i][j] = i \leq j$ 일 때 A_i 부터 A_j 까지의 행렬을 곱하는데 필요한 기본적인 곱셈의 최소 횟수 $1 \leq i \leq j \leq n$

$$= \begin{cases} \text{minimum}_{i \leq k \leq j-1} (M[i][k] + M[k+1][j] + d_{i-1}d_kd_j) & \text{if } i < j \\ M[i][i] = 0 \end{cases}$$

다음 테이블의 $M<2><5>$ 의 계산식을 다음과 같이 계산한다. 여기서 A, B를 하단의 빈 칸에 작성하라.

$M<2><5> = \text{MIN}\{15+8, 24+A+B, 42+12\}$

A는 위 식 $M<i><j>$ 를 구하는 식의 $M<k+1><j>$ 에 해당하고, B는 $d_{i-1}d_kd_j$ 에 해당한다. ecampus에서 괄호 문자는 별도의 기능으로 사용되므로 $< >$ 로 표시,

	1	2	3	4	5
1	0	24	42	60	35
2	0	0	24	42	$M[2][5]$
3	0	0	0	18	15
4	0	0	0	0	9
5	0	0	0	0	0

A=

B=

문제 11

10점

크기가 $n \times n$ 인 주어진 행렬 $A = \{a_{ij}\}$ 이 있다. $a_{ij} = 0$ for $j > i$. 행렬 A로부터 크기가 $n \times n$ 인 행렬 $S = \{s_{ij}\}$ 를 다음과 같이 계산한다.

For $p=1, n$, $s_{pk} = \sum_{r=1}^k a_{pr}$, 여기서 $k=1, p$. 그리고 $s_{ij} = 0$ for $j > i$

예를 들어, $n=3$ 일 때 다음과 같이 S를 만든다.

$$A = \begin{pmatrix} a_{11} & 0 & 0 \\ a_{21} & a_{22} & 0 \\ a_{31} & a_{32} & a_{33} \end{pmatrix}, S = \begin{pmatrix} s_{11} & 0 & 0 \\ s_{21} & s_{22} & 0 \\ s_{31} & s_{32} & s_{33} \end{pmatrix} = \begin{pmatrix} a_{11} & 0 & 0 \\ a_{21} & a_{21} + a_{22} & 0 \\ a_{31} & a_{31} + a_{32} & a_{31} + a_{32} + a_{33} \end{pmatrix}$$

A가 $n \times n$ 크기 일 때, S를 구축하는데 필요한 덧셈의 개수를 $T(n)$ 라 한다.

n 개의 원소를 더할 때 덧셈은 $n-1$ 회 수행한다고 가정한다.

$T(n)$ 을 n 의 함수로 표시하는 다음 식에서 계수 a, b, c, d 를 정수 숫자로 적으시오.

$$T(n) = \frac{1}{6} \times (a \times n^3 + b \times n^2 + c \times n + d)$$

빈칸에 정수 숫자로 답할 때 만일 답이 0 인 경우에도 반드시 0으로 입력해야 한다.

음수인 경우 -와 숫자를 붙여서 기입할 것.

(힌트) 먼저 다음의 식에서 괄호 안의 부분을 작성한 후 $T(n)$ 을 계산한다.

$$T(n) = \sum_{p=1}^n \sum_{k=1}^p (\dots)$$

a=

b=

c=

d=