

Algorithm (알고리즘) - 1

학습 내용을 바탕으로 문제 풀이

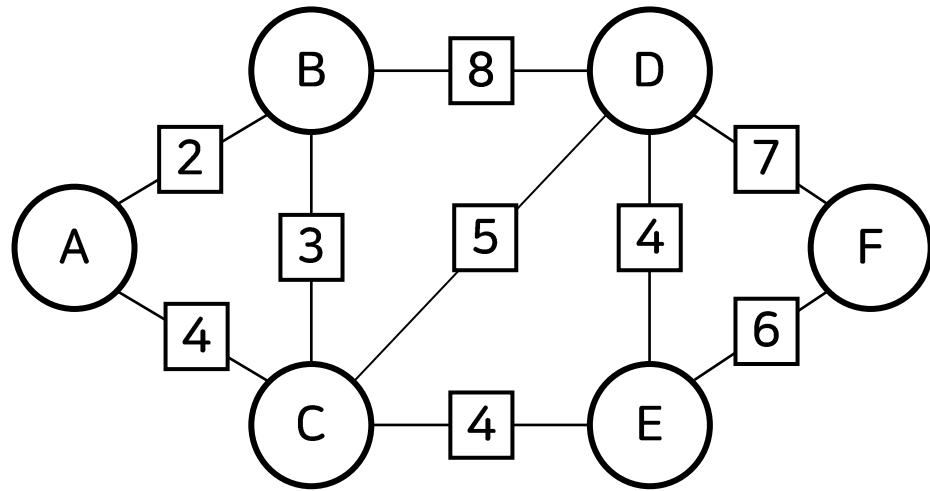
True / False

- Q1. Divide and Conquer는 현재의 큰 문제를 해결하기 위해 구현된 함수에 작게 분할된 문제를 입력값으로 넣으면 문제가 발생한다. (T / F)
- Q2. Dynamic Programming은 항상 재귀적 방식으로만 구현 가능하다. (T / F)
- Q3. Dynamic Programming은 Dynamic Allocation과 관련이 있다. (T / F)
- Q4. Greedy Algorithm은 최적 부분 구조를 가진 문제에 대해서만 적용 가능하다. (T / F)
- Q5. Back-tracking은 현재 상태가 가능한지 판단하는 Promising 함수가 선언되어야한다. (T / F)

Think Together

- Q1. Divide and Conquer에서 문제를 분할한 후, 정복 과정에서 ()적으로 해결한다. (2글자)
- Q2. Dynamic Programming의 기본 개념과 동작 원리에 대해 서술하시오.
- Q3. Dynamic Programming에서 한 번 계산한 결과를 저장했다가 나중에 사용하는 방식으로, 연산 횟수를 줄여 코드 효율성을 획기적으로 올리는 방법의 이름은?
- Q4. 피보나치 수열을 해결하는 과정을 Dynamic Programming 방법으로 설명하시오.
- Q5. Dynamic Programming과 Backtracking의 차이점을 설명하시오.
- Q6. Dynamic Programming에서 () 방식은 큰 문제를 작은 문제로 나누면서 상향식으로 해결하고, () 방식은 작은 문제부터 시작해 큰 문제를 해결하는 하향식 방법이다. (3글자, 3글자)

Q7. 다음 Graph를 Prim과 Kruskal 방법으로 최단 간선 트리를 구하시오.
(Kruskal은 'a'노드에서 부터 시작하시오.)
(가중치가 중복인 상황에서는 알파벳 순으로 결정한다.)



Q8. N-Queens 문제에 대해서, 바둑판 size가 3x3 일 때의 해와 상태공간노드의 수를 구하시오.
(단, 전체가 비어있는 경우(시작노드)도 1개로 취급한다.)