

아니요. 바로 요청하신 "탐색 문제들 풀이 구성"을 출력합니다. 아래 구성은 오직 제공 마크다운을 근거로 하며, 전략적 요소를 강조하고, 특별 처리는 작도(그리기) 부분에만 적용했습니다. 그 외 삽입·추가는 없습니다.

공통 풀이 프레임(전략 중심)

1. 선언

- 그래프 탐색으로 풀니다(이미 등장한 상태는 제외).
- 동점 규칙은 좌→우(또는 사전순)로 일관 적용합니다.

2. 초기화

- 루트=초기 상태, 깊이 0, 경로비용 $g=0$.
- 프린지 자료구조는 알고리즘별 규칙대로 준비.

3. 반복

- 프린지에서 선택 규칙에 맞는 노드를 꺼내 확장.
- 자식 생성 시 부모·깊이· g (필요 시 h, f) 갱신.
- 방문 집합에 따라 중복·재방문 차단.

4. 종료

- 목표 상태를 발견했을 때가 아니라, "꺼냈을 때 종료"가 필요한 알고리즘(UCS, A*)는 그 규칙을 따릅니다.
- 부모 포인터로 경로를 역추적해 최종 해를 명시.

5. 복잡도 기술

- 제공 문서의 표기만 사용: BFS/UCS $O(b^d)$, DFS $O(b^m)$, DLS/IDS $O(b l)$, Greedy/A* $O(b m)$.

작도(그리기) 특별 처리 체크리스트

- 노드 상자 표기: • BFS/DFS/DLS/IDS: 상태, 깊이 d . • UCS: 상태, $g(n)$. • Greedy: 상태, $h(n)$. • A*: 상태, $g(n)$, $h(n)$, $f(n)=g+h$.
- 간선에 비용이 있으면 간선 옆에 비용 라벨.
- 확장 순서 번호를 노드에 작게 표기.
- 매 단계 프린지 상태를 옆에 기입(Queue/Stack/PQ).
- 목표를 꺼낸 시점 표시 후, 부모 연결선에 굵은 표시로 최종 경로 강조.

알고리즘별 풀이 템플릿

1) BFS(너비우선)

- 자료구조: 큐.
- 초기화: $Q=[\text{시작}]$, 방문={시작}.
- 선택·확장: Q 의 앞에서 꺼내, 자식을 좌→우로 큐 뒤에 삽입. 새 상태만 방문·부모·깊이 기록.
- 종료: 목표 상태를 꺼냈을 때 경로 복원.

- 전략 포인트: 단위 비용일 때 깊이 기준 최적해.
- 복잡도(문서): 시간 $O(b^d)$, 공간 $O(b^d)$.
- 자주 하는 실수: 가중 그래프에도 "최단거리"로 단정, 방문 처리를 삽입 시점과 꺼낼 시점에 중복 적용.

2) UCS(균일비용)

- 자료구조: 우선순위큐(PQ), 키=g.
- 초기화: $PQ=[\text{시작}, g=0]$.
- 선택·확장: g가 가장 작은 노드를 꺼내 확장. 더 작은 g로 도달하면 PQ 갱신.
- 종료: 목표를 "꺼냈을 때" 종료(발견 시 종료 아님).
- 전략 포인트: 간선 비용 양수 가정하에 거리 기준 최적해 보장.
- 복잡도(문서): 시간 $O(b^d)$, 공간 $O(b^d)$.
- 자주 하는 실수: 목표 "발견" 시 조기 종료, 깊이 최적성과 혼동.

3) DFS(깊이우선)

- 자료구조: 스택.
- 초기화: $S=[\text{시작}]$.
- 선택·확장: 스택의 맨 위를 꺼내 자식을 역순으로 푸시하면 좌→우 확장 순서 유지.
- 종료: 목표를 만나면 경로 복원(단, 무한 가지·사이클 주의).
- 전략 포인트: 메모리 절약, 최적해 보장 없음.
- 복잡도(문서): 시간 $O(b^m)$, 공간 $O(b^m)$.
- 자주 하는 실수: 방문 집합 없이 무한 루프, "최단"으로 착각.

4) DLS(깊이제한)

- 자료구조: 스택, 최대 깊이 l.
- 초기화: 시작 깊이 0.
- 선택·확장: 깊이 < l까지만 DFS 확장, l에서 더 내려가지 않음.
- 종료: 목표가 깊이 ≤ l일 때 발견.
- 전략 포인트: l 설정을 해 깊이 이상으로.
- 복잡도(문서): 시간 $O(b^l)$, 공간 $O(b^l)$.
- 자주 하는 실수: 노드 깊이 계산 오류, l보다 깊은 목표를 찾았다고 오기.

5) IDS(반복적 깊이증가)

- 자료구조: l=0,1,2,...로 DLS 반복.
- 초기화: 각 l마다 새로 시작(이전 트리 폐기).
- 선택·확장: 매 레벨은 DLS와 동일.
- 종료: 목표 깊이에 도달한 l에서 발견.
- 전략 포인트: BFS의 깊이·최적성과 DFS 수준 메모리의 절충.
- 복잡도(문서): 시간 $O(b^l)$, 공간 $O(b^l)$.
- 자주 하는 실수: 이전 레벨 방문 집합을 다음 레벨로 넘김, 프린지를 누적.

6) Greedy Best-first

- 자료구조: 우선순위큐, 키=h.
- 초기화: h가 주어진 표로부터 각 노드에 할당.

- 선택·확장: h 가 가장 작은 노드부터 확장.
- 종료: 목표를 꺼냈을 때 경로 복원.
- 전략 포인트: 빠를 수 있으나 최적해 보장 없음.
- 복잡도(문서): 시간 $O(b^m)$, 공간 $O(b^m)$.
- 자주 하는 실수: A^* 와 혼동해 $f=g+h$ 로 표기, 사이클 제어 누락.

7) A^*

- 자료구조: 우선순위큐, $키=f=g+h$.
- 초기화: 시작 $g=0$, h 는 표로부터, $f=g+h$.
- 선택·확장: f 최소 노드를 꺼내 자식을 생성, $g'=g+cost$, h 는 표값, $f'=g'+h$.
- 종료: 목표를 "꺼냈을 때" 종료.
- 전략 포인트: h 는 과대평가 금지. f, g, h 를 모두 노드에 명시.
- 복잡도(문서): 시간 $O(b^m)$, 공간 $O(b^m)$.
- 자주 하는 실수: 목표 "발견" 시 종료, $g/h/f$ 중 일부 누락, h 과대평가.

답안 작성 템플릿(시험지에 그대로)

1. 규칙 선언: 그래프 탐색, 동점 규칙, 종료 규칙(UCS/A^* 는 꺼낼 때 종료).
2. 초기화: 프린지와 방문 집합, 시작 노드 표기(d 또는 $g/h/f$).
3. 단계별 전개: 각 단계의 프린지 상태와 확장 순서 번호 기입.
4. 종료·경로: 목표 꺼낸 시점, 부모 추적 경로, 총 비용 또는 깊이 명시.
5. 복잡도 및 성질: 해당 알고리즘의 완전성·최적성·복잡도를 문서 표기에 맞춰 한 줄로.

이 구성만 따라 그리면, 채점 포인트(프린지 기록, 재방문 차단, 꺼낼 때 종료, $g/h/f$ 표기, 동점 일관성)를 놓치지 않고 풀이를 일관되게 작성할 수 있습니다.