검정 안내

1. Sort 등의 기본적인 알고리즘을 reference로 참고할 수 있다.
2. Java library reference 도 존재한다.

S/W 검정 응시 주의사항

1. 신분증 지참 필수
2. 지각X
3. 퇴실은 검정시작 후 1시간 후부터 가능, 감독관에게 문의 (화장실도 1시간 후에)
4. scs 검정서버에 임시저장 수시로 하기
5. 마지막으로 제출된 코드로 평가됨!!!! -> 최고점 code 가 제출되는 것이 아님.
6. 추가시간은 pc문제등의 요소가 발생했을 때만
7. 최종결과는 1주일 후에 나온다.

사용가능 tool: visual studio 2017\_express, eclipse 2018-09

Pro 검정 난이도

* Adv의 경우 완전 탐색으로도 문제가 풀리지만 pro 단계에서는 무조건 fail이다!!!!
* 응시 요령: 분석/설계 1 + 구현 2 + 디버깅/최적화 1
* 문제 이해/출제자의 의도 파악 + 정확하고 빠른 구현 역량
* Sample testcase를 다 맞으면 합격인가요? -> no 라고 생각하자
* 절대평가(o) or 상대평가(x)

학습 범위

* 자료구조, 알고리즘, 시간 복잡도, 구현 숙련도(속도+정확도) 등

Visual studio에서는 잘 돌아가는데 검정 서버(gcc)에서는 에러가 나는 경우가 있다. gcc가 더 검증이 까다롭기 때문에 이런 경우가 생긴다.

1. Pro 검정 팁
   1. Stl 사용가능 but 직접 구현하는 게 빠를 수도 ㅇㅆ음
   2. 주어진 시간 내에 샘플 tc가 통과했다 해도 불합격 할 수 있음
   3. 전체 경우의 수가 큰 경우가 많다
      1. 완전 탐색, O(n^2) 이면 무조건 fail
      2. O(n), O(nlogn) 등을 잘 활용해라 ex. Linked list, hash, heap
   4. 문제 분석!!!
      1. 빨간색으로 강조한 부분 잘 읽을 것
      2. 제약사항 분석
      3. 함수 호출 횟수 제한 확인해라
   5. 구현
      1. 손코딩 먼저 하고 코딩하는 걸 추천
   6. 최적화 tip
      1. 함수 별 시간 check
      2. 함수 앞 부분에서 return을 시켜버리면 함수마다 걸리는 시간을 check 할 수 있으니 잘 이용해라
   7. 기본적인 자료구조, 알고리즘을 이해해야 한다.
   8. 100,000 이하인 조건이면 nlogn 정도가 나와야 한다.
   9. 문제는 1문제 4시간준다.
2. 구현 팁
   1. 초기화: {}중괄호 초기화 많이 이용해라
   2. 변수 초기화 주의!!!, 테스트 케이스마다 반복해서 사용하는 변수들 초기화 주의
   3. 메모리 크기를 넘지 않게 변수 선언(stack, heap 메모리 크기 주의, 재귀함수 쓰면 stack overflow 가능성 있으니 조심해라)
   4. Memory full vs 동적 할당에서 memory full을 주로 사용해라
   5. 가능한 STL 구현해서 사용해라
   6. 테스트 케이스 잘 돌아가면 code 개선해라!!! , N이 큰 케이스 여러 번 복사해서 돌려보는 것도 꿀 팁 이다.
   7. 생각하지 못한 부분에서 에러가 발생하진 않을 지 고민해봐라
   8. IDE 디버깅 툴 적극 활용
3. 접근법
   1. 문제가 어렵다 -> 어떻게든 먼저 해결부터 하자
   2. 문제가 쉽다 -> 시간 복잡도 계산 잘하자