

알고리즘 적용 기획서

광주 3반 고유창

광주 3반 김희주

1. 목차
2. 적용 알고리즘
 - 2-1) 이분탐색 (lowerBound, upperBound)
 - 2-1-1)알고리즘 적용 시 장점
 - 2-1-2)알고리즘 적용 시 단점
 - 2-2) 부분집합
 - 2-2-1)알고리즘 적용 시 장점
 - 2-2-2)알고리즘 적용 시 단점

2. 적용 알고리즘

2-1) 이분탐색

- 시나리오

사용자는 일정 수준 이상 혹은 이하의 금액대의 집을 찾고 싶어합니다. 예를들어 사용자는 100만원 이상에 해당하는 집을 찾고싶어하거나, 300만원 이하에 해당하는 집을 찾고싶어합니다. 이 때 원하는 금액대별을 입력하고 이상, 이하 버튼을 고른 후 검색버튼을 누르면 해당 금액대에 맞는 집을 조회하여 보여줍니다.

- 알고리즘 적용 방법

해당 검색옵션(등, 매입형태 등)에 맞는 모든 집을 배열또는 리스트로 저장한 뒤 거래금액을 기준으로 오름차순 정렬을 합니다. 이분탐색 알고리즘을 통해 해당금액 이상인 경우, 사용자가 원하는 금액대 중 최소 금액을 가진 HouseDeal의 index를 찾은 후 index부터 마지막 index까지의 HouseDeal만 반환합니다. 해당금액 이하인 경우, 사용자가 원하는 금액대 중 최대 금액을 가진 HouseDeal의 index를 찾은 후 0부터 index까지의 HouseDeal만 반환합니다.

- 적용 시 장점

이분탐색을 사용하지 않고 하나하나 금액대에 해당하는지 찾으면 시간복잡도는 $O(\text{HouseDeal 개수})$ 만큼이 걸립니다. 어플리케이션이 지속적으로 진행이 될 수록 저장되는 HouseDeal은 증가하게 됩니다. 즉, HouseDeal이 많아질수록 시간은 점점 길어지게 됩니다. 이분탐색을 사용하게 되면 검색범위를 반씩 줄여나가기 때문에 $O(\log N)$ 이 되고 금액대에 가장 가까운 금액의 인덱스만 알면 바로 원하는 데이터를 반환 할 수 있습니다.

- 적용 시 문제점

몇몇 데이터는 거래금액이 누락된 경우가 존재합니다. 이 누락의 의미가 거래금액이 0원인지 미정인지 판단할 수 없습니다. 하지만 정렬을 해야하기 때문에 임시로 0원으로 설정하였습니다. 만약 사용자가 이하 금액대를 보고싶으면 언제나 거래금액이 누락된 HouseDeal을 보게됩니다. 추후에 거래금액 미정 포함/미포함 옵션을 추가하면 정말 원하는 데이터만 보여 줄 수 있을 것 같습니다.

2-2)부분집합

- 시나리오

사용자가 보유한 자산을 토대로 최대한 몇 채의 집을 구매가능한지 알아보고 싶어합니다. 또한, 특정 전용면적 내의 집만 알아보고 싶을 때 부분집합 알고리즘을 사용하여 화면에 출력해 줍니다.

- 알고리즘 적용 방법

만약 100만원의 자산을 소유하고 있고 5채의 집을 구매하고싶어하면, 부분집합 알고리즘을 통해 5채의 집을 고른후 그 집들의 거래금액의 합이 100만원 이하이면 조건에 만족하므로 집들의 정보를 출력해줍니다.

- 적용 시 장점

일일이 스크롤을 내려서 원하는 조건에 맞는 데이터를 찾아봐야하는 수고를 덜수 있습니다.

- 적용 시 문제점

원하는 집의 종류를 선택하지 못하므로 만약 사용자가 아파트를 원할경우 결과로 주택이 나올수도 있다는 문제점이 있습니다. 이를 보완하기 위해서는 집에 대한 분류를 우선적으로 적용한 후에 부분집합 알고리즘을 적용해야 한다고 생각합니다.