

문제해결과 알고리즘

Binary Search

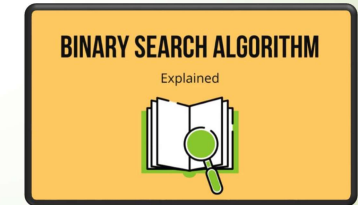
JEE, Jung Eun
rosaliejee@skku.edu

2

Contents

I. 탐색

II. 이진 탐색



3

1. 탐색 (1/3)

■ 탐색(Search)

- 컴퓨터에 저장한 자료 중에서 원하는 항목을 찾는 작업
 - 탐색 성공
 - 원하는 항목(값)을 찾은 경우
 - 탐색 실패
 - 원하는 항목(값)을 찾지 못한 경우
- 탐색 키를 가진 항목을 찾는 것
 - 탐색 키(search key)
 - 자료를 구별하여 인식할 수 있는 키
- 삽입/삭제 작업에서의 검색
 - 원소를 삽입하거나 삭제할 위치를 찾기 위해서 탐색 연산 수행

JEE, Jung Eun

4

1. 탐색 (2/3)

■ 탐색 방법

- 탐색 방식에 따른 분류
 - 비교 탐색 방식(comparison search method)
 - 탐색 대상의 키를 비교하여 탐색하는 방법
 - Ex. 순차 탐색, 이진 탐색, 트리 탐색
 - 계산 탐색 방식(non-comparison method)
 - 계수적인 성질을 이용한 계산으로 탐색하는 방법
 - Ex. 해싱
- ❖ 탐색 방법의 선택
 - 자료 구조의 형태와 자료의 배열 상태에 따라 최적의 탐색 방법 선택

JEE, Jung Eun

1. 탐색 (3/3)

■ 순차 탐색(sequential search)

- 선형 검색(linear search)
- 일렬로 된 자료를 처음부터 마지막까지 순서대로 탐색하는 방법
- 찾고자 하는 원소의 위치에 따라 비교 횟수가 결정됨
 - 평균 시간 복잡도 : $O(n)$

■ 이진 탐색(Binary Search, 이분 검색)

- 정렬되어 있는 자료에 대해서 수행하는 탐색 방법
- 삽입 or 삭제가 발생했을 경우에 항상 자료의 상태를 정렬 상태로 유지하는 추가적인 작업 필요함
 - 시간 복잡도 : $O(\log_2 n)$

2. 이진 탐색 (1/10)

■ 이진 탐색(Binary Search, 이분 검색)

- 자료의 **중간키 값**(가운데에 있는 값)을 **탐색키 값**과 비교를 통해 다음 탐색 위치를 결정하여 탐색을 수행하는 방법
 - 탐색 키(값) > 중간 키(값)
 - 중간 키보다 큰 오른쪽 부분에 대해서 탐색 실행
 - 탐색 키(값) < 중간 키(값)
 - 중간 키보다 작은 왼쪽 부분에 대해서 탐색 실행
- 탐색 키(값)를 찾을 때까지 이진 탐색을 순환적으로 반복 수행함으로써 탐색 범위를 반으로 줄여가면서 빠르게 탐색함
- 분할 정복 알고리즘 기법을 이용한 탐색 방법
 - 탐색 범위를 반으로 분할하는 작업과 탐색 작업을 반복 수행

2. 이진 탐색 (2/10)

■ 이진 탐색 알고리즘 핵심 내용

- $low \leq high$

$low (=0)$
↓

mid
↓
 $(=(low+high)/2)$

$high (=length-1)$
↓

Sorted data
- 중간 값(mid) 찾기
 - 중간 값(mid) 인덱스에 위치한 값이 탐색 키인가?
 - Yes → 위치(중간값 인덱스 + 1)반환
 - No → 탐색 계속
 - ✓ [선행:오름차순 정렬]
 - 1) 중간 값(mid)인덱스에 위치한 값 > 탐색키

$low (=0)$
↓

mid
↓

$high (=mid-1)$
↓
 - 2) 중간 값(mid)인덱스에 위치한 값 < 탐색키

$low (=mid+1)$
↓

mid
↓

$high (=length-1)$
↓

2. 이진 탐색 (3/10)

■ 이진 탐색 알고리즘 수행 과정

- ① 전체 리스트의 자료를 정렬(오름차순 or 내림차순)하여 저장한다.
- ② 탐색 키를 찾기 위해, 리스트 범위 내에서 작은 값(index)과 큰 값(index)을 설정한다.
- ③ 작은 값(index)과 큰 값(index)으로 중간 값(index)을 찾는다.
- ④ 탐색하고자 하는 탐색 키와 중간 값 인덱스에 위치한 값을 비교한다.
- ⑤ [오름차순 정렬일 경우] 탐색 키가 중간 값 인덱스에 위치한 값보다 크다면, 이후 찾을 범위에서 중간 값 왼쪽 부분을 제외하고, ②과정부터 반복 수행
- ⑥ [오름차순 정렬일 경우] 탐색 키가 중간 값 인덱스에 위치한 값보다 작다면, 이후 찾을 범위에서 중간 값 오른쪽 부분을 제외하고, ②과정부터 반복 수행
- ⑦ 탐색 키의 인덱스(index) 번호를 반환하여 문제를 해결한다.
 - 탐색 성공 : 탐색 키와 중간 값 인덱스에 위치한 값 일치
 - 탐색 실패 : 리스트에 탐색 키와 일치하는 값이 없음

2. 이진 탐색 (4/10)

■ [w10_Ex1] 이진 탐색 프로그램-반복문

■ 요구사항

- 1. 이진 탐색으로 입력 받는 탐색 키(search key)를 찾는 프로그램을 작성하시오.
 - 초기 데이터 : 43,97,13,96,84,51,64,25,72,14,93,33,6,95,53
 - 선행 작업 : 오름차순 정렬
- 2. 이진 탐색을 수행하는 반복문으로 구현한 사용자 정의 함수를 작성하시오.
- 3. 탐색을 성공하였을 경우, 몇 회에 찾았는지 출력 하시오.

■ 문제해결 및 알고리즘

- 이진 탐색 사용자 함수 정의 : 반복문 사용
- 메인코드
 - 이진 탐색 함수 호출
 - 함수 반환값(return value)으로 결과 출력
 - 탐색 성공 → 탐색 횟수, 위치(index) 출력
 - 탐색 실패

JEE, Jung-Eun

Week10

2. 이진 탐색 (5/10)

■ 이진 탐색 과정 Ex1. A = [43, 97, 13, 96, 84, 51, 64, 25, 72, 14, 93, 33, 6, 95, 53]

■ 선행 작업 : 데이터 정렬-오름차순

6	13	14	25	33	43	51	53	64	72	84	93	95	96	97
---	----	----	----	----	----	----	----	----	----	----	----	----	----	----

■ 특정 값 33을 이진 탐색 방법으로 찾기

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
6	13	14	25	33	43	51	53	64	72	84	93	95	96	97
low				mid		high								
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
6	13	14	25	33	43	51	53	64	72	84	93	95	96	97
low			mid		high									
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
6	13	14	25	33	43	51	53	64	72	84	93	95	96	97
low			mid		high									
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
6	13	14	25	33	43	51	53	64	72	84	93	95	96	97
low			mid		high									

➢ 탐색 횟수 : 4회

Week10

JEE, Jung-Eun

2. 이진 탐색 (6/10)

■ 이진 탐색 과정 Ex2. A = [43, 97, 13, 96, 84, 51, 64, 25, 72, 14, 93, 33, 6, 95, 53]

■ 선행 작업 : 데이터 정렬-오름차순

6	13	14	25	33	43	51	53	64	72	84	93	95	96	97
---	----	----	----	----	----	----	----	----	----	----	----	----	----	----

■ 특정 값 72를 이진 탐색 방법으로 찾기

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
6	13	14	25	33	43	51	53	64	72	84	93	95	96	97
low							mid							high
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
6	13	14	25	33	43	51	53	64	72	84	93	95	96	97
low								mid		high				
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
6	13	14	25	33	43	51	53	64	72	84	93	95	96	97
low								mid		high				

➢ 탐색 횟수 : 3회

JEE, Jung-Eun

Week10

2. 이진 탐색 (7/10)

■ [w10_Ex1] 실행 예시

```
[이진 탐색(오름차순정렬)]
원본 데이터: [43, 97, 13, 96, 84, 51, 64, 25, 72, 14, 93, 33, 6, 95, 53]
정렬 데이터: [6, 13, 14, 25, 33, 43, 51, 53, 64, 72, 84, 93, 95, 96, 97]
탐색 키(search key) 입력: 33
=>일치
=>4번에 탐색 성공!!!
=>탐색 위치(index)=4
```

```
[이진 탐색(오름차순정렬)]
원본 데이터: [43, 97, 13, 96, 84, 51, 64, 25, 72, 14, 93, 33, 6, 95, 53]
정렬 데이터: [6, 13, 14, 25, 33, 43, 51, 53, 64, 72, 84, 93, 95, 96, 97]
탐색 키(search key) 입력: 72
=>일치
=>3번에 탐색 성공!!!
=>탐색 위치(index)=9
```

```
[이진 탐색(오름차순정렬)]
원본 데이터: [43, 97, 13, 96, 84, 51, 64, 25, 72, 14, 93, 33, 6, 95, 53]
정렬 데이터: [6, 13, 14, 25, 33, 43, 51, 53, 64, 72, 84, 93, 95, 96, 97]
탐색 키(search key) 입력: 57
=>탐색 실패
```

Week10

JEE, Jung-Eun

2. 이진 탐색 (8/10)

■ [w10_Ex2] 이진 탐색 프로그램-재귀함수

■ 요구사항

1. 이진 탐색으로 입력 받는 탐색 키(search key)를 찾는 프로그램을 작성하시오.
 - 초기 데이터 : 43,97,13,96,84,51,64,25,72,14,93,33,6,95,53
 - 선행 작업 : 오름차순 정렬
2. 이진 탐색을 수행하는 재귀함수로 구현한 사용자 정의 함수를 작성하시오.
 - 단계별 탐색 과정에 대해 자세히 출력하시오.
3. 탐색을 성공하였을 경우, 몇 회에 찾았는지 출력하시오.

■ 문제해결 및 알고리즘

- 이진 탐색 사용자 함수 정의
 - 재귀함수 사용
 - 단계별 탐색 과정 출력
- 메인 코드
 - 이진 탐색 함수 호출 → 함수 반환값(return value)으로 결과 출력

JEE, Jung-Eun

2. 이진 탐색 (9/10)

■ [w10_Ex2] 실행 예시

[이진 탐색(오름차순정렬)]

원본 데이터: [43, 97, 13, 96, 84, 51, 64, 25, 72, 14, 93, 33, 6, 95, 53]
정렬된 데이터: [6, 13, 14, 25, 33, 43, 51, 53, 64, 72, 84, 93, 95, 96, 97]

탐색 키(search key) 입력:13
key=13, slist[7]=53 => 불일치
key=13, slist[3]=25 => 불일치
key=13, slist[1]=13 => 일치

=>3회에 탐색 성공!!!
=>탐색 위치(index)=1

[이진 탐색(오름차순정렬)]

원본 데이터: [43, 97, 13, 96, 84, 51, 64, 25, 72, 14, 93, 33, 6, 95, 53]
정렬된 데이터: [6, 13, 14, 25, 33, 43, 51, 53, 64, 72, 84, 93, 95, 96, 97]

탐색 키(search key) 입력:93
key=93, slist[7]=53 => 불일치
key=93, slist[11]=93 => 일치

=>2회에 탐색 성공!!!
=>탐색 위치(index)=11

[이진 탐색(오름차순정렬)]

원본 데이터: [43, 97, 13, 96, 84, 51, 64, 25, 72, 14, 93, 33, 6, 95, 53]
정렬된 데이터: [6, 13, 14, 25, 33, 43, 51, 53, 64, 72, 84, 93, 95, 96, 97]

탐색 키(search key) 입력:77
key=77, slist[7]=53 => 불일치
key=77, slist[11]=93 => 불일치
key=77, slist[9]=72 => 불일치
key=77, slist[10]=84 => 불일치

=>탐색 실패

JEE, Jung-Eun

2. 이진 탐색 (10/10)



■ [w10_Ex3] 과일 찾기 프로그램

■ 요구사항

1. 이진 탐색으로 입력 받는 과일을 리스트에서 찾는 프로그램을 작성하시오.
 - 선행 작업 : 내림차순 정렬
2. 10개의 과일로 리스트를 생성하시오.
3. 입력한 과일을 찾았다면 탐색 횟수와 저장 위치(index)를 출력하시오.
4. 무한 반복으로 과일을 찾을 수 있도록 하고, '끝' 입력 시 프로그램을 종료하시오.

■ 문제해결 및 알고리즘

- 이진 탐색 사용자 함수 정의
- 과일리스트 생성
 - list=[]
- 무한 반복으로 과일 찾기
 - while True

[실행 예시]

[과일 찾기 프로그램]

찾을 과일 입력 : 사과
사과를 1회 탐색으로 찾았습니다!!!
사과의 저장 위치(index)는 4입니다.

찾을 과일 입력 : 딸기
딸기를 3회 탐색으로 찾았습니다!!!
딸기의 저장 위치(index)는 8입니다.

찾을 과일 입력 : 수박
수박을 4회 탐색으로 찾았습니다!!!
수박의 저장 위치(index)는 3입니다.

찾을 과일 입력 : 자몽
자몽은(는) 과일 리스트에 없습니다.
찾을 과일 입력 : 바나나
바나나를 4회 탐색으로 찾았습니다!!!
바나나의 저장 위치(index)는 6입니다.

찾을 과일 입력 : 자두
자두는(는) 과일 리스트에 없습니다.

찾을 과일 입력 : 끝
프로그램을 종료합니다.

JEE, Jung-Eun

[W10] 도전과제 (1/2)



■ [w10_hw1] 서울 구별 '갤럭시Z폴드3' 재고현황 파악 프로그램을 작성하시오.

■ 요구사항

1. 삼성전자에서는 원활한 판매망 구축을 위해 성균관대 인사캠 주변 '갤럭시Z폴드3' 재고현황을 파악하고 있습니다.
 - 탐색 지역 : 종로구, 동대문구, 노원구, 성북구, 광진구, 서대문구, 중구
2. 2021년 10월 기준 각 구별 재고현황은 다음과 같다.
 - 재고 수량 : 종로구:33, 동대문구:21, 노원구:0, 성북구:29, 광진구:47, 서대문구:30, 중구:25
3. 사용자로부터 행정구역을 입력 받아, 탐색 지역 가능 여부를 출력하시오.
 - 탐색 지역이라면 재고 수량을 출력하시오.
4. 이진 탐색으로 탐색 지역 가능 여부를 탐색하는 사용자 정의 함수를 작성하시오.
 - 정렬 기준(오름차순, 내림차순)은 임의로 선택하고, 재귀함수를 사용하시오.
5. 실행 예시를 참고하여 프로그램을 완성하시오.
6. 임의의 변수 및 함수를 사용하고, 코드에 간단한 주석(설명)을 작성하시오.
7. 구름edu에서 테스트케이스 일치 여부를 확인한 후, 제출하시오.

- ❖ 본인 스스로 요구사항에 맞는 알고리즘을 구상해보고, 입력 데이터에 따른 올바른 결과가 출력되도록 코드를 작성하세요.

JEE, Jung-Eun

[W10] 도전과제 (2/2)

[w10_hw1 실행 예시]

[갤럭시Z폴드3 재고 현황]

행정구역 입력(종료:끝): 종료구
 행정구역은 탐색 지역입니다.
 행정구역에는 재고가 33대 있습니다.
 행정구역 입력(종료:끝): 광진구
 광진구는 탐색 지역입니다.
 광진구에는 재고가 47대 있습니다.
 행정구역 입력(종료:끝): 노원구
 노원구는 탐색 지역입니다.
 노원구에는 재고가 0대 있습니다.
 행정구역 입력(종료:끝): 동대문구
 동대문구는 탐색 지역입니다.
 동대문구에는 재고가 21대 있습니다.
 행정구역 입력(종료:끝): 성북구
 성북구는 탐색 지역입니다.
 성북구에는 재고가 29대 있습니다.
 행정구역 입력(종료:끝): 끝
 프로그램은 종료합니다.

[갤럭시Z폴드3 재고 현황]

행정구역 입력(종료:끝): 종료구
 행정구역은 탐색 지역입니다.
 행정구역에는 재고가 33대 있습니다.
 행정구역 입력(종료:끝): 성북구
 성북구는 탐색 지역입니다.
 성북구에는 재고가 29대 있습니다.
 행정구역 입력(종료:끝): 서대문구
 서대문구는 탐색 지역입니다.
 서대문구에는 재고가 30대 있습니다.
 행정구역 입력(종료:끝): 중구
 중구는 탐색 지역입니다.
 중구에는 재고가 25대 있습니다.
 행정구역 입력(종료:끝): 동대문구
 동대문구는 탐색 지역입니다.
 동대문구에는 재고가 21대 있습니다.
 행정구역 입력(종료:끝): 끝
 프로그램은 종료합니다.

[갤럭시Z폴드3 재고 현황]

행정구역 입력(종료:끝): 송파구
 송파구는 탐색 지역이 아닙니다.
 행정구역 입력(종료:끝): 광진구
 광진구는 탐색 지역입니다.
 광진구에는 재고가 47대 있습니다.
 행정구역 입력(종료:끝): 동작구
 동작구는 탐색 지역이 아닙니다.
 행정구역 입력(종료:끝): 노원구
 노원구는 탐색 지역입니다.
 노원구에는 재고가 0대 있습니다.
 행정구역 입력(종료:끝): 중구
 중구는 탐색 지역입니다.
 중구에는 재고가 25대 있습니다.
 행정구역 입력(종료:끝): 끝
 프로그램은 종료합니다.

JEE, Jung-Eun

Week10

[W10] 심화과제 (1/2)

➤ [w10_hw2] 이진 탐색으로 데이터를 탐색하는 과정을 출력하는 프로그램을 작성하시오.

➤ 요구사항

- 정렬 기준을 입력 받아 선행 작업을 수행하거나, 0을 입력 받아 프로그램을 종료 하시오.
 - 오름차순(1), 내림차순(2), 종료(0), 이외 입력에 대해서는 에러 처리하고 다시 입력 받기
- 정렬 기준에 따라 **이진 탐색을 수행**하는 사용자 정의 함수를 작성하시오.
 - 오름차순 정렬 데이터로 이진 탐색하는 사용자 정의 함수 작성
 - 내림차순 정렬 데이터로 이진 탐색하는 사용자 정의 함수 작성
 - 이진 탐색을 수행할 때, **단계별 탐색 과정**에 대해 출력 하시오.
- 탐색을 성공하였을 경우, **몇 회**에 찾았는지 출력 하시오.
- 이진 탐색을 **무한반복**으로 실행하고, 예시를 참고하여 프로그램을 완성 하시오.

❖ **본인 스스로 요구사항에 맞는 알고리즘을 구상해보고, 입력 데이터에 따른 올바른 결과가 출력되도록 코드를 작성하세요.**

JEE, Jung-Eun

[W10] 심화과제 (2/2)

[w10_hw2 실행 예시]

[이진 탐색]

오름차순(1), 내림차순(2), 종료(0): 1
 탐색 키(search key) 입력: 33
 key=33, slist[7]=53 => 불일치
 key=33, slist[3]=25 => 불일치
 key=33, slist[5]=43 => 불일치
 key=33, slist[4]=33 => 일치
 => 4번에 탐색 성공!!!
 => 탐색 위치(index)=4
 오름차순(1), 내림차순(2), 종료(0): 2
 탐색 키(search key) 입력: 33
 key=33, slist[7]=53 => 불일치
 key=33, slist[11]=25 => 불일치
 key=33, slist[9]=43 => 불일치
 key=33, slist[10]=33 => 일치
 => 4번에 탐색 성공!!!
 => 탐색 위치(index)=10
 오름차순(1), 내림차순(2), 종료(0): 3
 입력 오류, 다시입력하세요.

오름차순(1), 내림차순(2), 종료(0): 1
 탐색 키(search key) 입력: 72
 key=72, slist[7]=53 => 불일치
 key=72, slist[11]=93 => 불일치
 key=72, slist[9]=72 => 일치
 => 3번에 탐색 성공!!!
 => 탐색 위치(index)=9
 오름차순(1), 내림차순(2), 종료(0): 2
 탐색 키(search key) 입력: 72
 key=72, slist[7]=53 => 불일치
 key=72, slist[3]=93 => 불일치
 key=72, slist[5]=72 => 일치
 => 3번에 탐색 성공!!!
 => 탐색 위치(index)=5
 오름차순(1), 내림차순(2), 종료(0): 0

JEE, Jung-Eun



JEE, Jung-Eun