

# 자료구조와 알고리즘

## HW1

2021706066 김나래

## 학번 짝수: 백준 1920번 문제 (강의자료 02-24)

### 1. Linear Search

#### 1. IDE의 코드 및 실행결과 창 (강의자료 02-30과 유사한 형태로 캡처)

##### A. IDE 코드

```
1920.py
1 # 백준 1920
2
3 import sys
4 input = sys.stdin.readline
5 sys.stdin = open('C:\\Marae\\2025\\25-1\\자료구조와알고리즘\\V41\\1920.txt', 'r')
6 n = int(input())
7 card = list(map(int, input().split()))
8 card.sort()
9 t = int(input())
10 target = list(map(int, input().split()))
11
12 linear_results = []
13 def linear_search(t, card):
14     for i in card:
15         if i == t:
16             return 1
17     return 0
18 for i in range(t):
19     t = target[i]
20     linear_results.append(linear_search(t, card))
21 for r in linear_results:
22     print(r)
23
24
c:\Marae\2025\25-1\자료구조와알고리즘\V41>
c:\Marae\2025\25-1\자료구조와알고리즘\V41>
c:\Marae\2025\25-1\자료구조와알고리즘\V41> c: && cd c:\Marae\2025\25-1\자료구조와알고리즘\V41 && cmd /c "c:\Program Files\Python311\python.exe" c:\Users\USER\vscode\extensions\ms-python.debugpy-2025.4.1-win32-x64\bundle\libs\debugpy\launcher 56194 -- c:\Marae\2025\25-1\자료구조와알고리즘\V41\1920.py
1
0
0
1
```

##### B. 실행 결과 창

```
c:\Marae\2025\25-1\자료구조와알고리즘\V41>
c:\Marae\2025\25-1\자료구조와알고리즘\V41>
c:\Marae\2025\25-1\자료구조와알고리즘\V41> c: && cd c:\Marae\2025\25-1\자료구조와알고리즘\V41 && cmd /c "c:\Program Files\Python311\python.exe" c:\Users\USER\vscode\extensions\ms-python.debugpy-2025.4.1-win32-x64\bundle\libs\debugpy\launcher 56186 -- c:\Marae\2025\25-1\자료구조와알고리즘\V41\1920.py
1
1
0
0
1
```

### 2. 코드 제출 결과 (강의자료 02-31)

BAE<JOON>  
ONLINE JUDGE

문제 문제집 대회 6 채점 현황 랭킹 게시판 그룹 더 보기

1920번 제출 맞힌 사람 슯코딩 재채점 결과 채점 현황 **내 제출** 질문 게시판

[NEW] 코딩 테스트 준비 온라인 강의 **활인 중** / 코딩 테스트 및 인터뷰 온라인 강의

1920 fort3354 모든 언어 모든 결과 검색

제출 번호	아이디	문제	결과	메모리	시간	언어	코드 길이	제출한 시간
91922131	fort3354	1920	시간 초과			Python 3 / 수정	418 B	24초 전

### 3. 소스코드 (TEXT OR 캡처)

- ## 백준 1920 번
- 
- import sys

```

4. import bisect
5.
6. sys.stdin = open('C:\\Narae\\kw\\2025\\25-
  1\\자료구조와알고리즘\\HW1\\1920.txt', 'r')
7.
8. N = int(input()) # list 크기
9. card = list(map(int, input().split()))
10. card.sort() # list 정렬
11. M = int(input()) # target 크기
12. target = list(map(int, input().split()))
13.
14. # 1. linear Search
15. linear_results = []
16. def linear_search(t, card):
17.     for i in card:
18.         if i == t:
19.             return 1
20.
21.     return 0
22. for i in range(M):
23.     t = target[i]
24.     linear_results.append(linear_search(t, card))
25. for r in linear_results:
26.     print(r)
27.

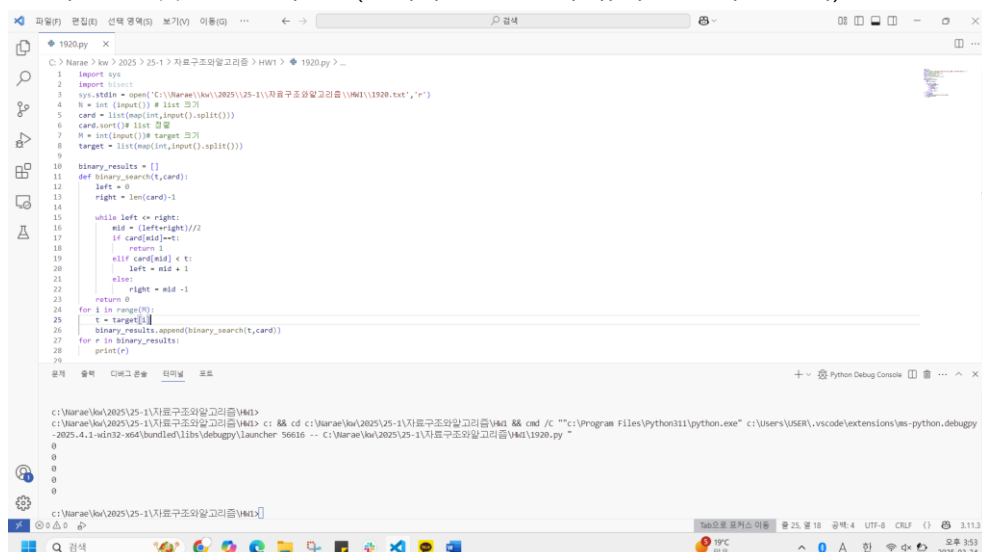
```

#### 4. 결과 혹은 코드에 대한 분석이나 의견

Linear Search로 진행된 수 찾기 문제는 시간 복잡도가  $O(N*M)$ 으로  $N$ 과  $M$ 이 커질 경우 반복 횟수가 매우 많아져, 주어진 시간 제한 내에 해결이 어려워질 수 있다.

## 2. Binary Search

A. IDE의 코드 및 실행결과 창 (강의자료 02-30과 유사한 형태로 캡처)



```

문제 출력 디버그 콘솔 터미널 포트
Python Debug Console

c:\Narae\kw\2025\25-1\자료구조와알고리즘\HW1>
c:\Narae\kw\2025\25-1\자료구조와알고리즘\HW1>
c:\Narae\kw\2025\25-1\자료구조와알고리즘\HW1> c: && cd c:\Narae\kw\2025\25-1\자료구조와알고리즘\HW1 && cmd /c ""c:\Program Files\Python311\python.exe" c:\Users\USER\vscode\extensions\ms-python.debugpy-2025.4.1-win32-x64\bundle\libs\debugpy\launcher 56194 -- C:\Narae\kw\2025\25-1\자료구조와알고리즘\HW1\1920.py ""
1
1
0
0
1

c:\Narae\kw\2025\25-1\자료구조와알고리즘\HW1>

```

## B. 코드 제출 결과 (강의자료 02-31)

BAE<K>JOON

ONLINE JUDGE

fort3354 | 설정 | 로그아웃

문제 문제집 대회 채점 현황

랭킹

게시판

그룹

더 보기

internship

전액무료 | 부트캠프

이스트소프트 KDT 인턴십 프로그램

EST

BACKEND

1920번

제출

맞힌 사람

숏코딩

재제점 결과

채점 현황

내 제출

질문 게시판

[NEW] 코딩 테스트 준비 온라인 강의 **할인 중** / 코딩 테스트 및 인터뷰 온라인 강의

1920

fort3354

모든 언어

모든 결과

검색

제출 번호	아이디	문제	결과	메모리	시간	언어	코드 길이	제출한 시간
91922276	fort3354	1920	맞았습니다!!	48616 KB	440 ms	Python 3 / 수정	589 B	38초 전

## C. 소스코드 (TEXT OR 캡처)

```

2. ## 백준 1920 번
3.
4. import sys
5. import bisect
6.
7. sys.stdin = open('C:\\Narae\\kw\\2025\\25-1\\자료구조와알고리즘\\HW1\\1920.txt', 'r')
8.
9. N = int(input()) # list 크기
10. card = list(map(int, input().split()))
11. card.sort() # list 정렬
12. M = int(input()) # target 크기
13. target = list(map(int, input().split()))
14.
15. #2. binary search
16. binary_results = []
17. def binary_search(t, card):
18.     left = 0
19.     right = len(card)-1
20.
21.     while left <= right:
22.         mid = (left+right)//2
23.         if card[mid]==t:
24.             return 1
25.         elif card[mid] < t:
26.             left = mid + 1

```

```
27.         else:
28.             right = mid - 1
29.         return 0
30.
31. for i in range(M):
32.     t = target[i]
33.     binary_results.append(binary_search(t, card))
34. for r in binary_results:
35.     print(r)
```

- A. 결과 혹은 코드에 대한 분석이나 의견
- i. Binary Search로 진행된 수 찾기 문제는 시간 복잡도가  $O(M \times \log N)$ 으로  $N$ ,  $M$ 이 큰 값이어도 시간제한인 1초 내에 충분히 계산 가능하다.