

## 과제 4

장유선

2023.08.07

### 1. 문제 정의

검색 기능을 사용할 때는 항상 문자열의 전체를 이용하여 검색을 수행하지 않는다.

예를 들어 "판교역 신분당선"을 검색하는 경우 "판교역", "판교" 등등으로 입력이 가능하다.

이를 구현하기 위해서는 단순히 각 문자열에 대하여 검색을 수행하는 방법이 존재하지만, 입력 데이터의 개수가 늘어나는 경우에는 검색 속도가 매우 느려질 수 있다.

이를 위하여 문자열을 아래와 같이 분리하여 저장한 데이터를 이용하여 빠르게 검색을 수행할 수 있다.

"판교역신분당선"

-> "판", "교", "역", "신", "분", "당", "선"

-> "판교", "교역", "역신", "신분", "분당", "당선"

-> "판교역", "교역신", "역신분", "신분당", "분당선"

-> "판교역신", "교역신분", "역신분당", "신분당선"

-> "판교역신분", "교역신분당", "역신분당선"

-> "판교역신분당", "교역신분당선"

-> "판교역신분당선",

#### <입력데이터>

입력 데이터는 아래와 같이 구성됩니다.

- 검색을 수행할 명칭 (1줄)
- 검색을 수행할 명칭과 검색 대상을 각각 한줄 씩 사용한다.

분당
판교역신분당선
판교역장항선
정자역분당선

### <출력데이터>

- 검색된 결과 데이터를 output.txt 로 출력

판교역신분당선

정자역분당선

## 2. Python Code Hard Copy

```
def split_input(input_data):
    index = {}
    for line in input_data:
        for i in range(len(line)):
            for j in range(i + 1, len(line) + 1):
                substring = line[i:j]

                if substring not in index:
                    index[substring] = []
                index[substring].append(line)
    return index

def search(input_lines):
    search_target = input_lines[0]
    index = split_input(input_lines[1:])

    found_strings = set()
    if search_target in index:
        found_strings.update(index[search_target])

    return found_strings

input_data = []
while True:
    line = input("")
    if not line:
        break
    input_data.append(line)

search_results = search(input_data)
with open("output.txt", "w") as output_file:
    for result in search_results:
        output_file.write(result + "\n")
```

### 3. Code 분석

<함수 설명>

1.

```
def split_input(input_data):
    index = {}
    for line in input_data:
        for i in range(len(line)):
            for j in range(i + 1, len(line) + 1):
                substring = line[i:j]

                if substring not in index:
                    index[substring] = []
                index[substring].append(line)

    return index
```

Input\_data의 모든 부분 문자열을 생성하고 Input\_data는 key로, 모든 부분 문자열은 value로 저장한다.

2.

```
def search(input_lines):
    search_target = input_lines[0]
    index = split_input(input_lines[1:])

    found_strings = set()
    if search_target in index:
        found_strings.update(index[search_target])

    return found_strings
```

input에서 첫줄은 검색대상으로 설정한다. 나머지는 split\_input 함수를 통해 인덱스를 생성한다. 검색대상이 인덱스에 존재한다면 해당 키의 값을 결과에 추가한다.

3.

```
input_data = []
while True:
    line = input("")
    if not line:
        break
    input_data.append(line)

search_results = search(input_data)
```

```
with open("output.txt", "w") as output_file:
    for result in search_results:
        output_file.write(result + "\n")
```

빈줄을 입력하면 종료하도록 입력을 받고, search함수를 호출하여 검색을 한 후, 결과를 output.txt 파일에 저장한다.

#### 4. 입력

분당

판교역신분당선

판교역장항선

정자역분당선

```
분당
판교역신분당선
판교역장항선
정자역분당선
```

#### 5. 결과

판교역신분당선

정자역분당선

