

[과제 4] 연속 행렬 곱셈

- 알고리즘과 문제해결 -

1분반 20180018 강 승아

[문제]

1. MatrixChain 알고리즘을 Python code로 작성하시오. (hw4.py)
2. 최소곱셈을 위한 행렬곱의 순서를 출력하도록 Python code를 update 하시오. (hw4a.py)

1. 구현사항

- 사용 언어 : Python
- 구현 내용 :

(1) 라이브러리

```
import sys
```

sys 라이브러리를 호출한 뒤, 정수 무한 값 초기화를 위한 sys.maxsize 함수를 사용하였다.

(2) 입력 값

```
d = [10, 20, 5, 15, 30]
n = len(d)
C = [[0 for col in range(n)] for row in range(n)]
```

행렬의 각 행과 열값을 중복 없이 d라는 리스트에 입력하였다. 행렬의 길이를 n에 할당하였으며 곱셈 횟수를 기록하기 위한 2차원 리스트를 모두 0의 값으로 할당 해주었다.

(3) 곱셈 횟수 계산

```
for L in range(1, n):
    for i in range(1, n - L):
        j = i + L
        C[i][j] = sys.maxsize
        for k in range(i, j):
            temp = C[i][k] + C[k + 1][j] + (d[i - 1] * d[k] * d[j])
            if temp < C[i][j]:
                S = k
                C[i][j] = temp
```

부분 문제를 추려내기 위해 L이라는 인덱스를 사용하였다. 이후 i행 j열(i보다 L만큼 큰) 값을 부분 문제에서 최대화 시킨다. 최대화는 비교되지 않기 위한 ∞ 값을 만들어 줌을 의미한다. 부분 문제를 반복하며 if문을 통해 만약 temp에 입력된 새로운 값이 기존 최소 횟수보다 적을 경우에 최소 횟수를 변경할 수 있게 하였다. S는 몇 번째의 반복에서 최솟값을 찾았는가를 기억하기 위한 변수이다.

(4) 곱셈 순서 계산 함수

```
def sequence(s):  
    if s == 0:  
        print("곱셈 순서는 A*(B*C*D)가 이상적입니다.")  
    elif s == 1:  
        print("곱셈 순서는 (A*B)*(C*D)가 이상적입니다.")  
    elif s == 2:  
        print("곱셈 순서는 (A*B*C)*D가 이상적입니다.")
```

위의 S값에서 저장된 최종 값을 통해 곱셈순서를 유추하였고 그에 따라 이상적 곱셈 순서를 제시하는 함수이다. 그러나 이는 4개의 행렬에서만 곱셈 순서를 제시할 수 있다.

2. 체크포인트 (1-(2)의 입력 값에 따른 실행 화면)

```
C:\Users\강승아\PycharmProjects\Matr  
최소 곱셈 횟수 : 4750  
  
Process finished with exit code 0
```

[hw4.py의 실행결과]

```
C:\Users\강승아\PycharmProjects\Matr  
최소 곱셈 횟수 : 4750  
곱셈 순서는 (A*B*C)*D가 이상적입니다.  
  
Process finished with exit code 0
```

[hw4.py의 실행결과]

3. 결론 및 느낀 점

수업을 들으며 이론은 비교적 쉽게 이해한 반면 구현할 때에 어려웠던 점은 단연 반복문이 여러 차례 반복되어 나오던 indexing 이었다. 부분 문제로 분할하기도 하며 역삼각형 형태로 채워지는 배열 이기에 필연적으로 반복문이 여러 개 필요했는데 구현 과정에서 강의자료의 알고리즘과 index가 맞지 않아 곤란했었다. 몇 시간을 할애하며 아쉬운 마음이 컸다. 알고리즘을 직접 구현하며 하나씩 배열을 출력해보기도 했고 이와 같이 코딩을 직접 해보는 것은 이번 수업에서 특히 도움이 많이 된 것 같다. 또한 같은 행렬의 곱이더라도 곱셈 횟수가 n배씩 차이가 나는 것을 보며 효율적 알고리즘의 필요성도 직접 확인 할 수 있었다.

4. 참고 문헌

- 금오공과대학교 컴퓨터시스템보안 8주차 연속 행렬 곱셈1, 2 강의 (최태영)
- 최대 정수 값 - <https://developeryuseon.tistory.com/47>