알고리즘 멘토링

- Dynamic Programming –

- 주민찬 -

Dynamic Programming

- 문제를 여러 하위 문제로 분할하고, 하위 문제들의 해결 결과를 저장하며, 이를 활용하여 더 큰 문제의 해를 효율적으로 계산하는 방법

1. Bottom – Up
2. Top – Down (Memoization)

Dynamic Programming

- 대표적인 예시 피보나치 수열

```python
def fibo(n):
    if n == 0 or n == 1:
        return n
    return fibo(n-1) + fibo(n-2)
```

```
1  fibo(10)
✓ 0.0s

55
```

```
1  fibo(40)
✓ 39.0s

102334155
```

Bottom – Up Dynamic Programming

```python
def fibo_bottom_up(n):
    fib = [0, 1] + [0] * (n)
    for i in range(2,n+1):
        fib[i] = fib[i-1] + fib[i-2]
    return fib[n]
```

```
1  fibo_bottom_up(10)
✓ 0.0s

55
```

```
1  fibo_bottom_up(40)
✓ 0.0s

102334155
```

Top – Down Dynamic Programming

```python
def fibo_top_down(n, memoization = {}):
    if n in memoization:
        return memoization[n]

    if n == 0 or n == 1:
        memoization[n] = n
    else:
        memoization[n] = fibo_top_down(n - 1, memoization) + fibo_top_down(n - 2, memoization)

    return memoization[n]
```

```
1  fibo_top_down(10)
✓ 0.0s

55
```

```
1  fibo_top_down(40)
✓ 0.0s

102334155
```

Dynamic Programming

- 대표적인 예시

1. 피보나치 수열
2. 최단 거리 탐색
3. 최장 증가 부분 수열 (LIS, Longest Increasing Subsequence)
4. 최장 공통 부분 수열 (LCS, Longest Common Subsequence)
5. 배낭 문제 (Knapsack Problem)


실습!!