



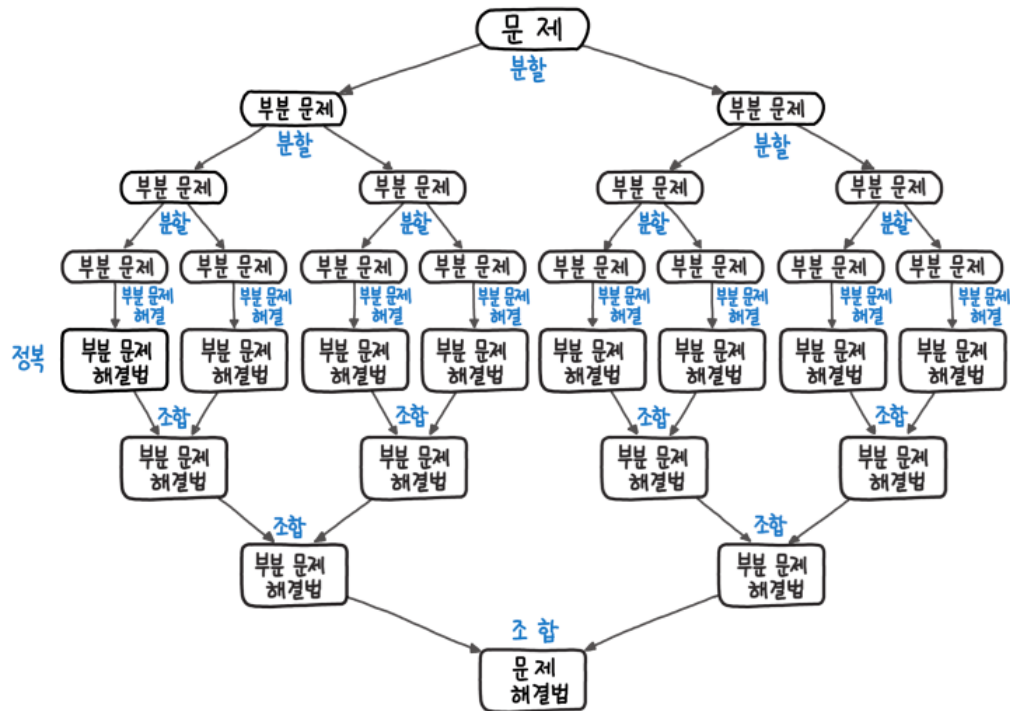
# 분할정복

[1. 분할 정복이란](#)

[2. 관련 문제 풀이 \(백준 1629 곱셈\)](#)

## 1. 분할 정복이란

- 분할정복이란, 문제를 나눌 수 없을 때까지 나누어서 각각을 풀면서 다시 합병하여 문제의 답을 얻는 알고리즘이다. 분할 정복은 말 그대로 ‘분할’해서 ‘정복’한 다음 정답을 ‘조합’해 나간다는 의미를 지닌다.
- 분할정복은 재귀를 활용하는 대표적인 알고리즘이기도 하다.
  - 대표적인 예시 : 병합정렬, 퀵정렬, 이진탐색 등
- 분할 정복 설계 방식
  - Divide(분할) : 문제를 동일한 유형의 여러 하위 문제로 나눈다.
  - Conquer(정복) : 가장 작은 단위의 하위 문제를 해결하여 정복한다. (재귀적으로 해결)
  - Combine(조합) : 하위 문제에 대한 결과를 원래 문제에 대한 결과로 조합한다.



출처: 파이썬 알고리즘 인터뷰

- 분할정복과 DP의 차이점 : **부분 문제의 중복**
  - 분할 정복은 분할된 하위 문제가 동일하게 중복이 일어나지 않는 경우에 쓰며, 동일한 중복이 일어나면 동적 프로그래밍을 활용한다.
- <https://www.notion.so/DP-54a18d3a54314d26aaf58199f7ddbb35#9dae0e070c154d20b210982f6a1659f3>

## 2. 관련 문제 풀이 (백준 1629 곱셈)

- 자연수 A를 B번 곱한 수를 알고 싶다. 단 구하려는 수가 매우 커질 수 있으므로 이를 C로 나눈 나머지를 구하는 프로그램을 작성하시오.
- 입력
  - 첫째 줄에 A, B, C가 빈 칸을 사이에 두고 순서대로 주어진다. A, B, C는 모두 2,147,483,647 이하의 자연수이다.

- 예제 입력
  - 2 10 12
  - 4

```
# 1629 곱셈

import sys

# a : 곱하는 수, b : 곱하는 횟수
def f(a, b):
    if b == 1:
        return a % C

    # a = 2, b = 10, 11
    # 2^10 = 2^5 * 2^5, 2^11 = 2^5 * 2^5 * 2

    else:
        tmp = f(a, b // 2)

        # b가 짝수인 경우
        if b % 2 == 0:
            return (tmp * tmp) % C

        # b가 홀수인 경우
        else:
            return (tmp * tmp * a) % C

A, B, C = map(int, sys.stdin.readline().split())

print(f(A,B))
```

a	b	return	retrun 순서
2	10	$(8 * 8) \% 12 = 4$	4
2	5	$(4 * 4 * 2) \% 12 = 8$	3
2	2	$(2 * 2) \% 12 = 4$	2
2	1	2	1

공통 문제 : 백준 5904 Moo 게임

개인 문제 : 분할정복 개인문제 모음