C++ & Algorithm Seminar

5. 재귀 함수

SHIN Hyun (kyaryunha)

한양대학교 컴퓨터소프트웨어학부 19학번 전기전자컴퓨터 학술 동아리 바라미 27기

May 20, 2022



Contents

1 Explanation - 재귀 함수

- 2 Problem 재귀 함수
 - BOJ 2747 피보나치 수



Explanation - 재귀 함수



재귀 함수라?

재귀 함수란

■ 자신을 다시 호출하는 함수 / 아래는 a 부터 b 까지의 합을 출력하는 함수

```
#include<iostream>
using namespace std;
int sum(int a, int b) { /// 재귀 함수
    if(a==b) return b; /// a 가 b랑 같으면 b 를 리턴하고
    return a + sum(a+1, b); /// 아니라면 a + sum(a+1, b) 를 리턴
int main(){
    int a, b;
    cin>>a>>b;
    cout << sum (a, b);
```

Problem - 재귀 함수



BOJ 2747 피보나치 수

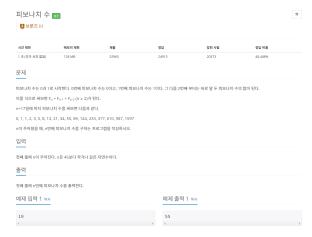


Figure: BOJ 2747 피보나치 수 (https://www.acmicpc.net/problem/2747)



코드는 다음과 같다. (이것은 시간복잡도를 설명하기 위한 예시일 뿐, 너무 오래 걸리기 때문에 답이 아니다)

```
#include<iostream>
using namespace std;
int fibo(int N) {
    if(N==1 || N==2) return 1;
    return fibo(N-1) + fibo(N-2);
int main(){
    int N:
    cin>>N;
    cout << fibo(N);
```

시간복잡도라?

- 시간 복잡도는 문제를 해결하는데 걸리는 시간과 입력의 함수 관계
- 예를 들어 for 문을 N번 돌리면 O(N) 이런식으로?
- 앞의 코드를 보면, 한 번 호출시, 자기 자신을 2번 호출하므로, 2^{N-1} + 2^{N-2} 번 코드를 실행한다. (이런 경우 차수가 큰 것 기준으로, 계수를 떼고, O(2^N) 이라 한다.
- 문제에서 N 이 45까지라 했는데, 이는 (대충 보면) 2⁴⁵ 라 생각하면 된다.
- 컴퓨터는 대략 1초에 1억 10억 (컴파일러가 얼마나 최적화를 잘했냐에 따라 다르긴함) 정도의 계산을 한다 생각하면 된다.
- 따라서 앞의 코드는 시간 초과다.



여담으로 총 계산량을 (생각하지 않고) 세고 싶으면 다음과 같이 하면 된다.

```
#include<iostream>
using namespace std;
long long cnt = 0;
int fibo(int N) {
    cnt++;
    if(N==1 || N==2) return 1;
    return fibo(N-1) + fibo(N-2);
int main(){
    int N;
    /// cin>>N;
    cout << fibo (45) << '\n';
    cout << cnt << '\n'; /// 2269806339
```

DP 메모이제이션을 적용한 코드는 다음과 같다. (이건 말로 설명할 예정)

```
#include<iostream>
using namespace std;
int dp[50];
int fibo(int N) {
    if(N==1 || N==2) return 1;
    if(dp[N]) return dp[N];
    else return dp[N] = fibo(N-1) + fibo(N-2);
int main(){
    int N;
    cin>>N;
    cout << fibo(N) << ' n';
```