

피보나치 수열 특정 번지의 값을 구할 때 재귀함수와 반복문을 이용할 수 있는데

코드 작성 후 테스트 중 두 방법에 속도 차이가 발생한다는 것을 알게 되었다.

오늘 수업 시간에 배운 시간 복잡도라는 개념이 떠올랐고, 각각의 수행시간을 한번 계산해보았다.

| <pre>int RecursiveFunction(const int n) { //재귀함수 if (n <= 1) return n; return RecursiveFunction(n - 1) + RecursiveFunction(n - 2); }</pre> | | | |
|---|-----|----------------------|----------------------|
| 함수 내부 행 번호 | s/e | 빈도 | 단계 수 |
| 1 | 1 | $(N^2 - N) / 2 + 2$ | $(N^2 - N) / 2 + 2$ |
| 2 | 1 | $(N^2 - 3N) / 2 + 2$ | $(N^2 - 3N) / 2 + 2$ |

| <pre>int RepetitiveFunction(const int n) { //반복문 if (n <= 1) return n; int now = 1; int prev = 0; int temp; for (int i = 2; i <= n; i += 1) { temp = now; now += prev; prev = temp; } return now; }</pre> | | | |
|---|-----|-------|-------|
| 함수 내부 행 번호 | s/e | 빈도 | 단계 수 |
| 1 | 1 | 1 | 1 |
| 2 | 1 | 1 | 1 |
| 3 | 1 | 1 | 1 |
| 4 | 1 | 1 | 1 |
| 5 | 1 | N | N |
| 6 | 1 | N - 1 | N - 1 |
| 7 | 1 | N - 1 | N - 1 |
| 8 | 1 | N - 1 | N - 1 |
| 9 | 0 | 0 | 0 |
| 10 | 1 | 1 | 1 |

재귀함수는 총 단계 수가 $N^2 - 2N + 2$, 반복문은 총 단계 수가 $4N + 2$ 로 나왔다. ($N > 1$ 기준)

내 계산이 어느 정도 틀렸을 수도 있지만 어차피 정밀한 계산을 하려면 step per execution이란 요소를 저수준까지 끌고 갈 정도의 이해가 있어야 한다고 생각해서 별 의미를 두지 않기로 했다.

아무튼 빅-오 표기법으로 재귀함수의 시간 복잡도는 $O(N^2)$ 이고 반복문의 시간 복잡도는 $O(N)$ 이어서 매개 변수의 값이 커질 때 두 방법의 시간 복잡도 차이도 커진다는 결론을 도출했다.