

Recursive Function

알린이월드 - 알형

What is recursive function?

1. **Recursive function** is a technique where a function calls itself.

→ 재귀는 함수가 자기 자신을 호출하는 기술입니다.

2. **Recursive function** can be a bit hard to understand.

→ 재귀는 약간 이해하기 어려울 수 있습니다.

3. **The best way to learn it is by experimenting.**

→ 재귀를 배우는 가장 좋은 방법은 직접 실험해보는 것입니다.

마트료시카 (러시아 전통 인형)



Summary

- 1) 재귀 함수는 자기 자신을 호출하는 함수입니다. (A recursive function is a function that calls itself.)
- 2) 큰 문제를 더 작은 문제로 나누고, 그 작은 문제들을 같은 방식으로 해결합니다.(It breaks a big problem into smaller problems and solves the smaller ones in the same way.)
- 3) 가장 작은 문제에 도달하면 직접 해결하고, 그 답을 가지고 더 큰 문제를 해결합니다.(When it reaches the smallest problem, it solves it directly and then returns the answer to solve the bigger problem.)

재귀함수 구현 방법

1. **종료 조건(Base Case, termination condition)**: 재귀를 멈추는 조건을 명확히 정의
2. **작은 문제로 나누기**: 문제를 더 작은 하위 문제로 쪼개기
3. **재귀 호출과 결과 병합**: 작은 문제를 해결하고 결과를 결합

1. 스택(Stack)이란?

- 스택은 '접시를 쌓아 올리는 것'과 비슷한 구조입니다.
- LIFO(Last In, First Out) 방식을 따릅니다
 - 제일 나중에 쌓은(위에 올린) 데이터가 제일 먼저 빠져나옵니다.
 - 즉, 맨 위에 있는 것만 뺄 수 있는 구조입니다.

2. 함수 호출 시 스택이 어떻게 사용될까?

(1) 함수 호출 시 “스택 프레임”이 쌓인다.

- 프로그램에서 함수를 부르면, 새로운 스택 프레임(Stack Frame)이 **스택 위**에 올려집니다.
- 스택 프레임에는 해당 함수가 동작하는 데 필요한 정보들이 들어 있어요.
 1. 매개변수 (함수로 넘어온 인자)
 2. 지역 변수 (함수 내부에서 선언된 변수)
 3. 복귀 주소 (함수가 끝나면 어디로 돌아가야 하는지)
- 함수가 끝날 때는 스택 맨 위의 스택 프레임을 제거(팝)하고, 그 아래에 있던 함수의 위치로 돌아갑니다(복귀).

(2) 함수가 끝나면 스택에서 제거된다.

- 함수가 **return** 을 만나거나 끝까지 실행을 마치면,
 1. 스택의 맨 위 스택 프레임을 꺼냅니다(pop).
 2. 복귀 주소에 따라 호출했던 함수(혹은 메인 영역)로 돌아갑니다.

