

# 배우는 내용

1. 프로그래밍언어 개요
2. 구문 구조-언어가 제공하는 의미와 형식 개요
3. 변수-속성, 바인딩, 타입검사..
4. 타입-문자열, 배열, 포인터, 구조적 타입도
5. 제어구조-수식, assignment, 반복문
6. 부프로그램, argument 구조적 언어
7. 객체, 추상데이터타입
8. 함수형 (functional) 언어
9. 논리적 (logical)언어 새로운 개념

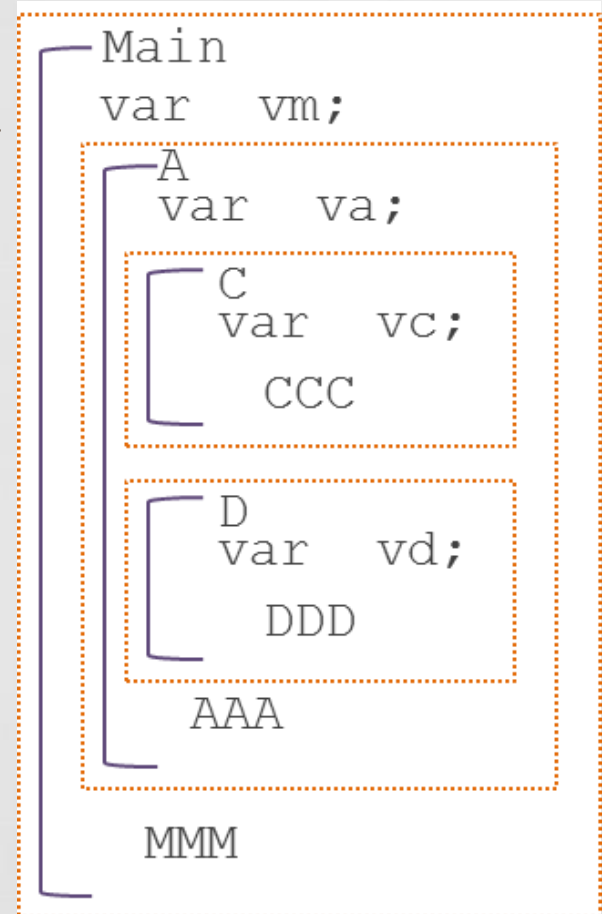
# 4장 영역 (Scope)

오른쪽에 주어진 프로그램 골격에서, block D의 DDD 지점을 실행할 때, 사용 가능한 변수들을 모두 나열하고 있는 것은?

(힌트: 모든 box는 외부에서는 내부가 보이지 않지만, 내부에서는 외부가 보인다고 생각해서, 각 block에서 보이는 변수를 선택하면 된다.)

- (a) vm, va, vc
- (b) vd
- (c) va, vc, vd
- (d) vm, va, vd

*C? Java? C++? ...*



# 변수의 Scope

- 변수 scope의 종류
  - 비지역변수(nonlocal variables)
    - 전역변수 (global variables)
    - **more : 지역도 아니지만 전역도 아닌 변수**
  - 지역변수 (local)
- scope rule(영역 규칙)
  - 프로그램의 어떤 지점에서 사용된 변수 이름을 어디에서 선언된 변수와 대응시켜줄 것인지를 결정하는 규칙
  - **static scoping rule (SSR)** : 수행 전에 중첩구조로 변수 대응
  - **dynamic scoping rule (DSR)** : 수행 중에 호출 체인으로 변수 대응

- C, PHP, Python의 예...

- python의 예

```
day = "Monday"
```

```
def foo();
```

```
    global day
```

```
    print (day)           # 출력: Monday
```

```
    day = "Tuesday"
```

```
    print (day)          # 출력: Tuesday
```

- Scope은 공간적(spatial) 개념이고, 존속기간은 시간적(temporal) 개념

- 완전 별개

eg) `void f() { static int count = 0; ...}`

- 용어: referencing environment