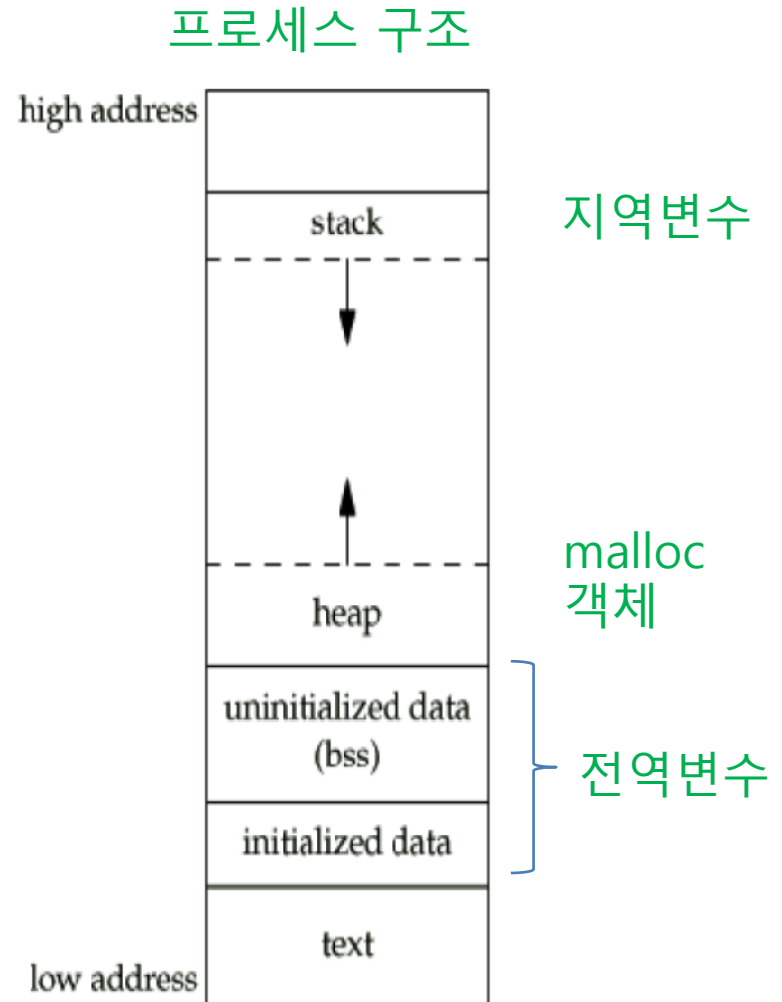


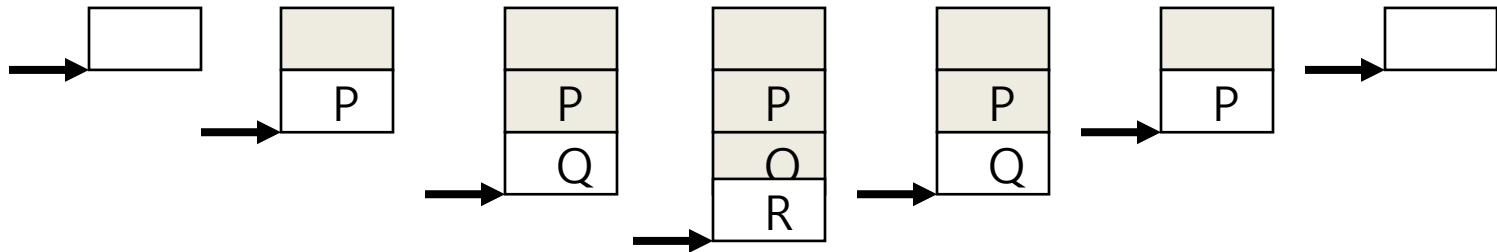
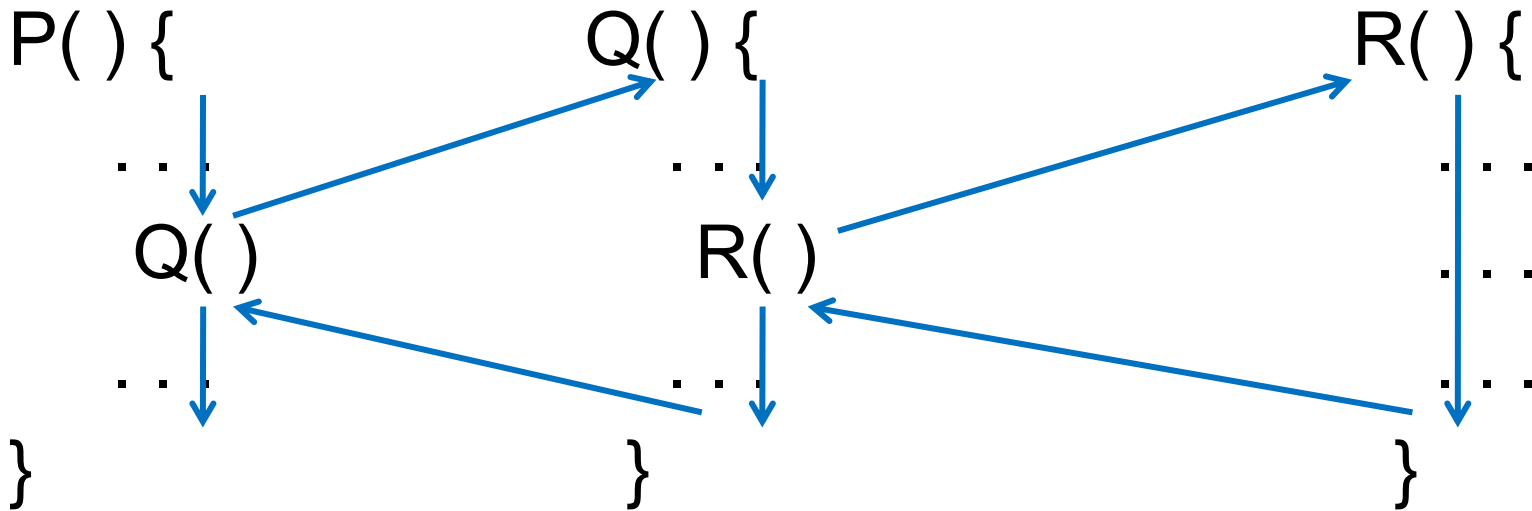
함수 호출 시 스택의 모습 (call-by-value)

변수들의 종류 및 메모리 공간 할당 시기

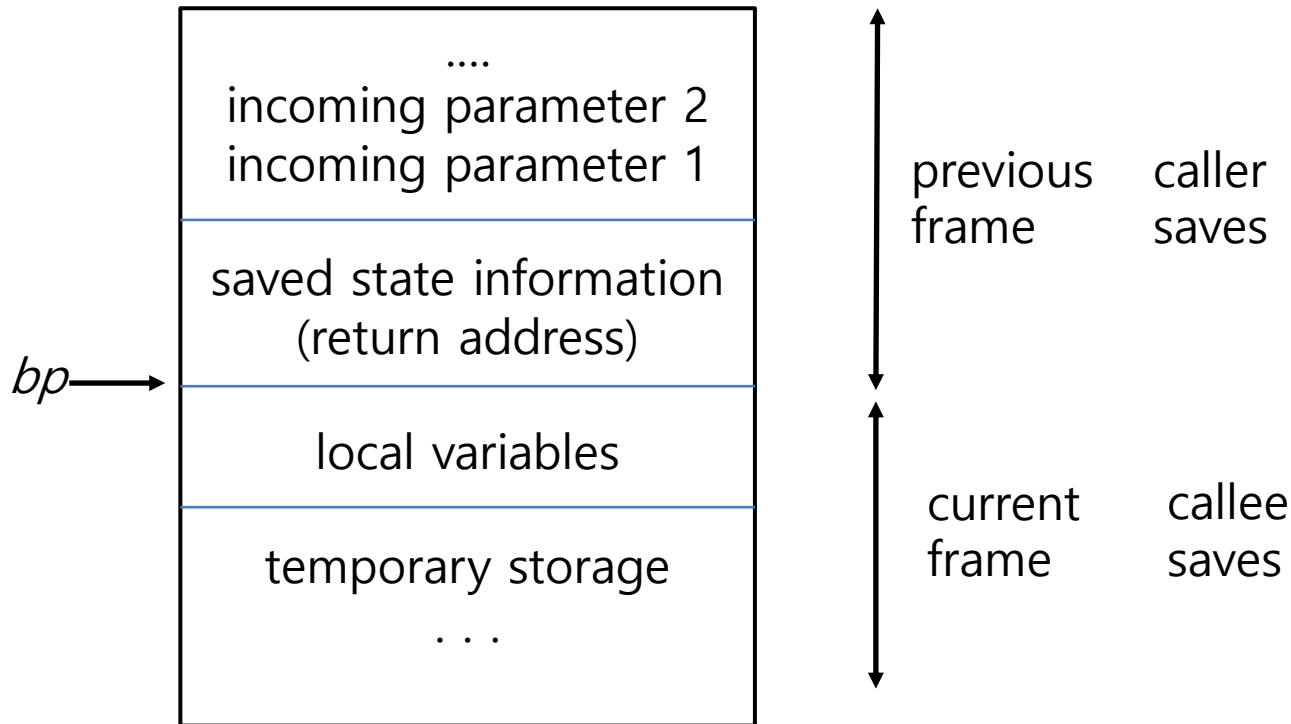
- 전역(global) 변수
 - 프로그램 전체에 사용됨
 - 프로세스 시작 때부터 공간 배당
- 지역(local) 변수
 - 오직 함수 내에서만 효과를 가짐
 - 동적 할당 : 함수 호출 시 할당되고, return에 회수
 - Stack에 할당됨
- malloc 등에 의해 할당되는 공간
 - malloc 함수 호출 시 동적으로 할당
 - Heap에 할당됨
 - 공간회수를 위해 free 함수 이용
- OOP의 객체
 - `Class a = new Class()`
 - malloc과 같은 동적 할당
 - Automatic Garbage Collection (자동메모리 관리 기능) 으로 free 불필요



스택의 동적인 변환



스택 프레임 구조



- 스택의 접근은 bp (혹은 fp, frame point)를 기준으로, 상대 주소 (offset이라고 부름)로 이루어 짐.

함수 호출 및 복귀 시의 스택 변화

- 함수 호출시
 - 파라미터의 값을 계산하여, 그것을 저장함 (call-by-value)
 - 호출되는 함수의 수행 후, 돌아올 주소(return address)를 저장함
 - 예전 fp (frame point) 를 저장하고, 호출되는 함수의 스택 주소를 fp를 할당함.
- 호출된 함수에서 할당되는 공간
 - 함수 수행 전에, 그 함수의 지역 변수(local variables)를 위한 공간 할당
 - 그 함수 수행 과정에서 임시로 필요한 공간을 할당함
 - 함수 수행을 종료 하면, fp의 값에 old fp를 할당하고
 - pc (program counter)에 저장되었던 return address를 할당함
 - return value는 지정된 register (예를 들어, register 0) 에 할당함