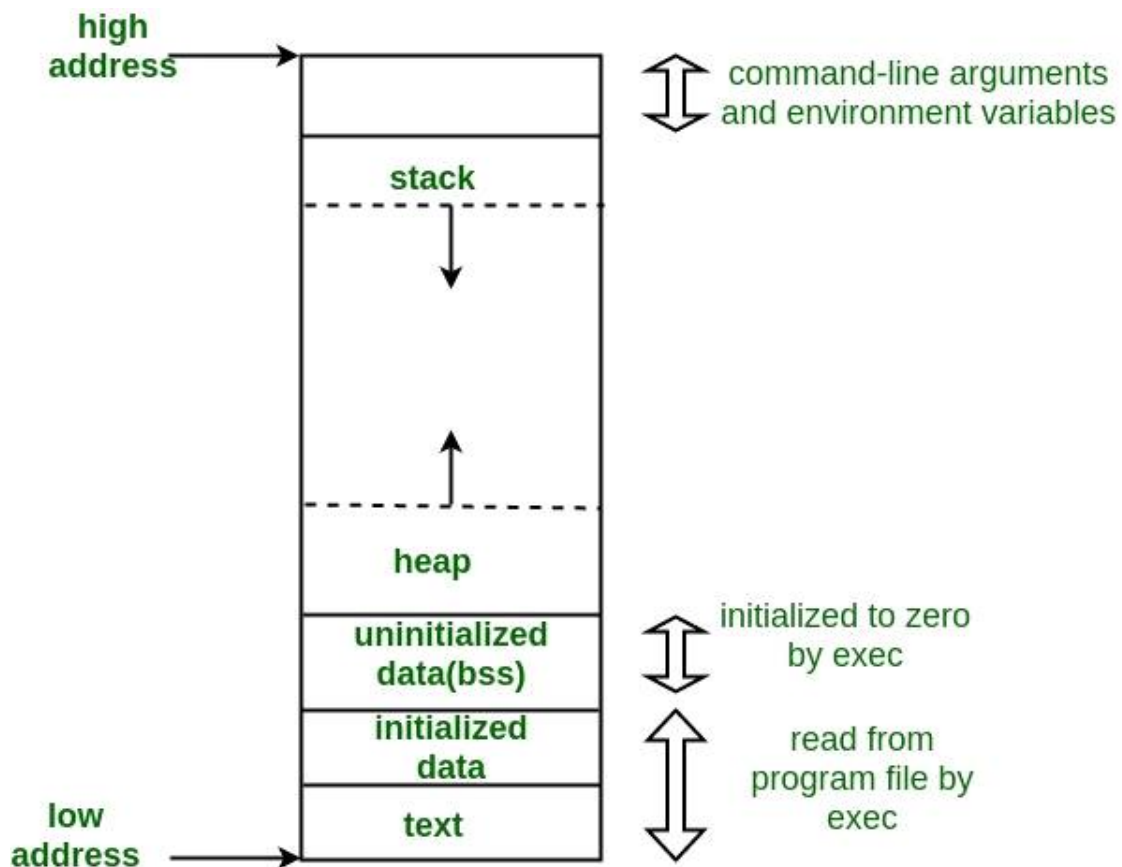
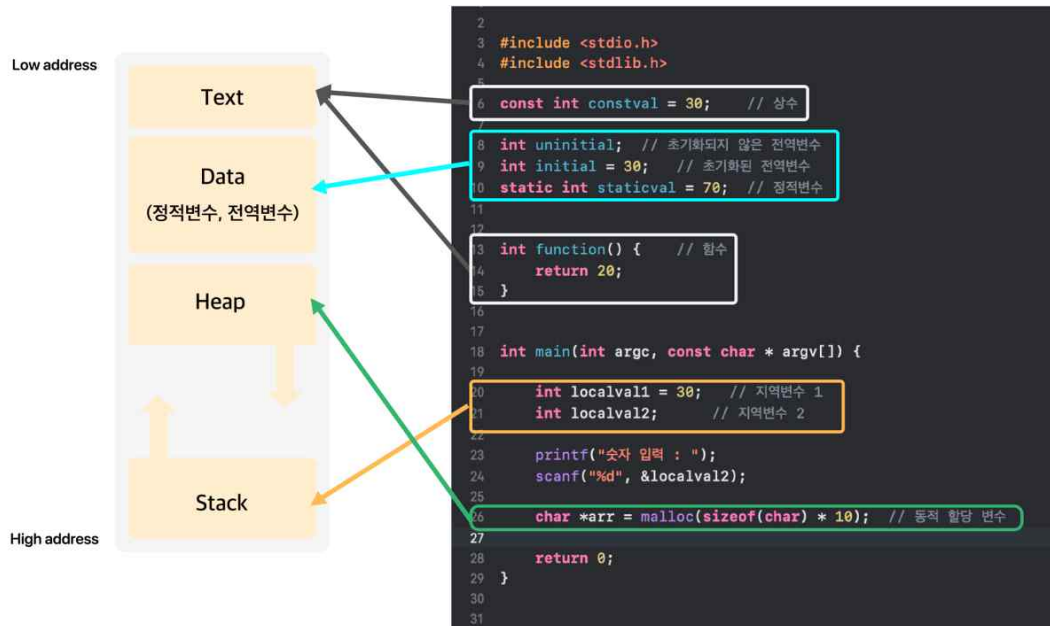


본인의 경우 2번째 그림인 Stack을 위에다가 놓는것이 더 옳다고 생각함.  
(스택은 높은 주소에서 낮은 주소인 아래방향으로 쌓이는 방식이므로)



- address = 0xFFFFFFFF

1. Stack : 함수에 대한 정보, 지역변수에 대한 정보

: (높은주소에서 낮은 주소 방향으로 메모리가 저장된다.)

2. Heap : 동적으로 할당되는 변수가 저장된다.

: (낮은 주소에서 높은 주소 방향으로 메모리가 저장된다.)

3. BSS : 프로그램에서 사용될 변수의 실제 위치 : 변수가 초기화가 되지 않았다. ex  
: int uninitial;

4. Data : 초기화된 변수가 저장 ex : int initial =30;

: 전역(global), 정적(static) 변수, 배열(array), 구조체(structure)가 저장되는 영역으로 초기화된 데이터가 저장된다.

5. Text or Code: assembly language가 저장되는 곳 : 시스템이 알아들을 수 있도록 컴파일러가 만든 실질적 코드가 저장됨.

: (overwrite를 방지하기위해 heap과 stack 아래 주소에 위치한다.). 즉 여기에는 함수, 제어문, 상수(const int) 영역 등이 저장된다.

: CPU가 코드 영역에 저장된 명령어를 하나씩 가져가서 처리하게 되며 컴파일 타임에 결정되고 중간에 코드를 바꿀 수 없게 read-only로 지정된다.

- address = 0x00000000