

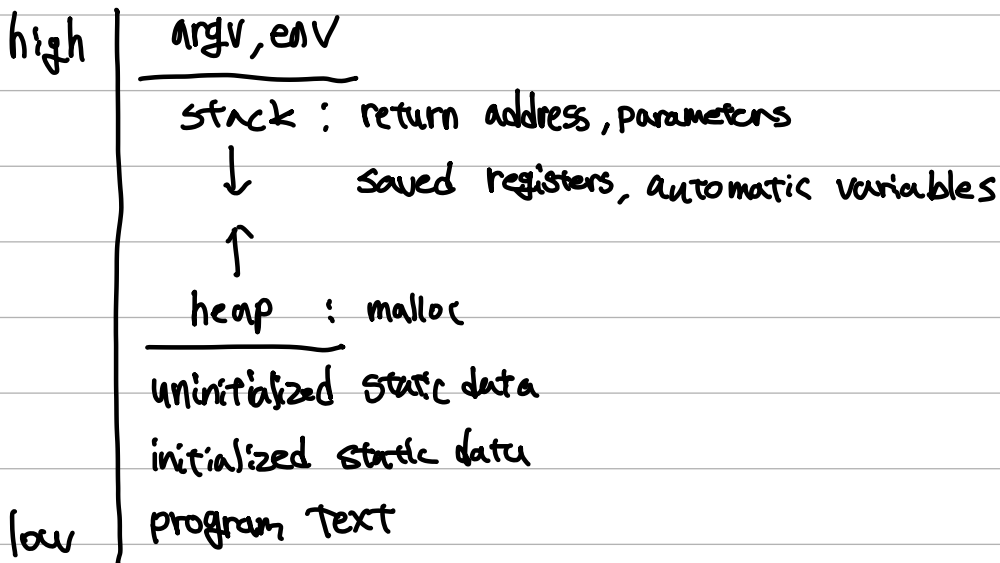
## Chapter 2.

Program v.s. Process  
↓  
실행중인 프로그램

Thread - own execution stack, program counter, register set, state.

Multi processes v.s. Multi threads  
↓  
낮은 오버헤드  
추가적인 동기화 필요

### Program image.



`void perror (const char *s)` → 현재 `errno` 에 해당하는 에러메시지 출력

`char* strerror (int errnum)` → 에러메시지를 가리키는 포인터 반환

## Thread-safe 예시

- strtok → 내부에 토큰화할 문자열 원본 자체를 저장하여 문제!

- char\* strtok\_r (char\* s, const char\* sep, char\*\* lastp)
  - 문자열
  - 파싱할 구분자
  - 파싱한 시작지점  
인자로 주어 해결