1. A stack frame, also known as an activation record, is the area of stack which stores subprogram call data such as the return address, passed arguments, local variables, and register information.
2. Calling Convention
3. The book examples always start functions with "push ebp" and "mov ebp,esp" because they are part of an assembly language implementation of the ability to initialize and access parameters during a function call in higher level languages. It creates a new stack frame within the callee, while the stack frame of the caller is preserved. “push ebp” saves the value of the EBP register, while “mov ebp, esp” points EBP to the top of the stack.
4. push rbp

mov rbp,rsp

1. Microsoft x64 calling convention:

* Followed on Windows and pre-boot UEFI
* First four integer or pointer arguments passed in registers RCX, RDX, R8, R9.

XMM0, XMM1, XMM2, XMM3 are used for floating point arguments.

* Floating point return values stored in XMM0
* 32 bytes of “shadow space” must be pushed onto the stack before calling function, and to be popped afterward.

System V AMD64 ABI calling convention

* Followed on Solaris, Linux, FreeBSD, macOS, and Unix
* First six integer or pointer arguments passed in registers RDI, RSI, RDX, RCX, R8, R9 (R10 used for static chain pointer).

XMM0, XMM1, XMM2, XMM3, XMM4, XMM5, XMM6, XMM7 are used for floating point arguments.

* Floating point return values stored in XMM0 and XMM1
* No shadow space