```
push rbp
mov  rbp,rsp
mov esi,6
mov edi,5
call AFunc
add rsp,8
```

高地址

↓

低地址

栈的分布	
rbp,rsp →	old_rbp

栈底

寄存器	
rax	
rbx	
rcx	
rdx	
rdi	
rsi	

PS: 执行语句之前的RBP在此栈空间的更高处

```
push rbp
mov  rbp,rsp
mov esi,6
mov edi,5
call AFunc
add rsp,8
```

栈的分布	
执行前的rsp →	old_rbp

栈底

寄存器	
rsi	0x6

```
push rbp
mov  rbp,rsp
mov esi,6
mov edi,5
call AFunc
add rsp,8
```

栈的分布	
执行前的rsp →	old_rbp

栈底

寄存器	
rdi	0x5
rsi	0x6

```
push rbp
mov  rbp,rsp
mov esi,6
mov edi,5
call AFunc
```

rdi:0x5 第一个参数

rsi:0x6 第二个参数

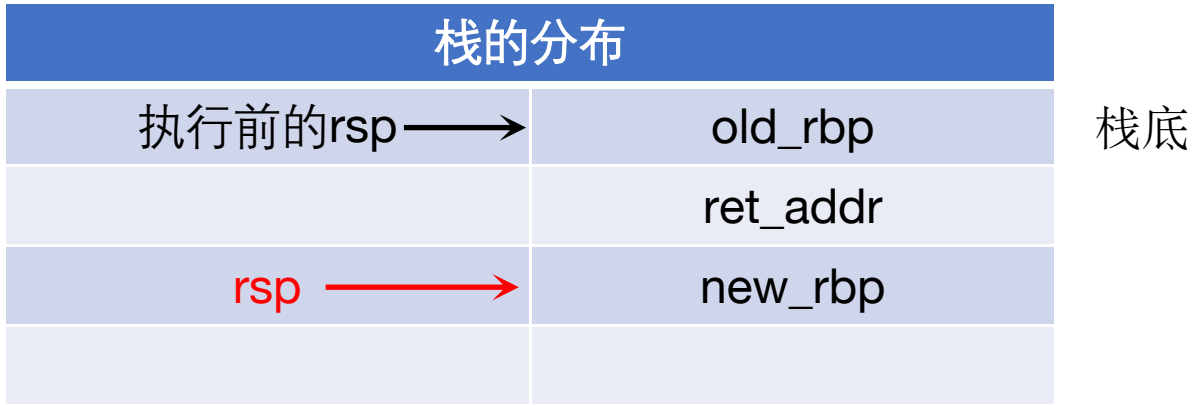
```
add rsp,8
```

栈的分布	
执行前的rsp →	old_rbp
rsp →	ret_addr

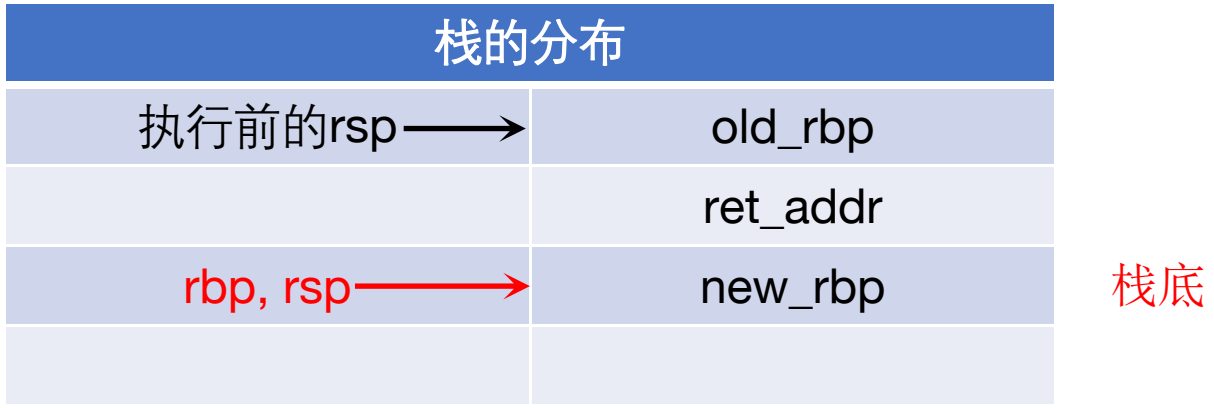
栈底

AFunc

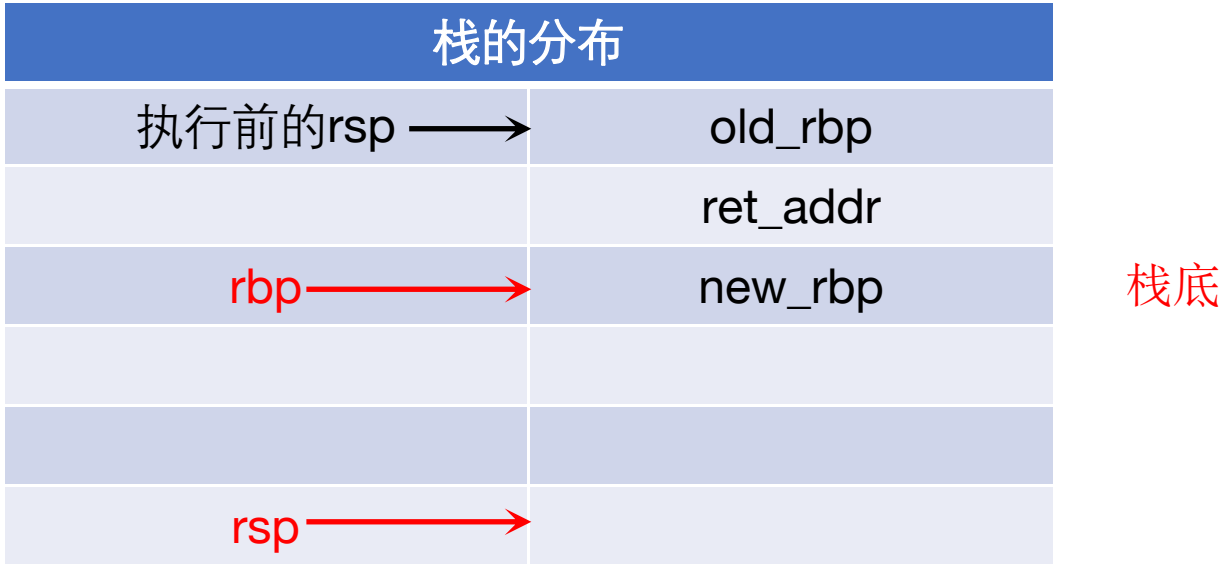
```
push rbp
mov rbp,rsp
sub rsp,0x18
mov DWORD PTR [rbp-0x14],edi
mov DWORD PTR [rbp-0x18],esi
mov DWORD PTR [rbp-0x4],0x3
mov DWORD PTR [rbp-0x8],0x4
mov eax, DWORD PTR [rbp+0x14]
mov DWORD PTR [rbp-0x4],eax
mov    eax, DWORD PTR [rbp - 0x18]
mov DWORD PTR [rbp-0x8],eax
mov    edx, DWORD PTR [rbp - 8]
mov    eax, DWORD PTR [rbp - 4]
mov esi,edx
mov edi,eax
call BFunc
mov eax,8
leave
ret
```



```
AFunc
push rbp
mov rbp,rsp
sub rsp,0x18
mov DWORD PTR [rbp-0x14],edi
mov DWORD PTR [rbp-0x18],esi
mov DWORD PTR [rbp-0x4],0x3
mov DWORD PTR [rbp-0x8],0x4
mov eax, DWORD PTR [rbp+0x14]
mov DWORD PTR [rbp-0x4],eax
mov    eax, DWORD PTR [rbp - 0x18]
mov DWORD PTR [rbp-0x8],eax
mov    edx, DWORD PTR [rbp - 8]
mov    eax, DWORD PTR [rbp - 4]
mov esi,edx
mov edi,eax
call BFunc
mov eax,8
leave
ret
```



```
AFunc
push rbp
mov rbp,rsp
sub rsp,0x18
mov DWORD PTR [rbp-0x14],edi
mov DWORD PTR [rbp-0x18],esi
mov DWORD PTR [rbp-0x4],0x3
mov DWORD PTR [rbp-0x8],0x4
mov eax, DWORD PTR [rbp+0x14]
mov DWORD PTR [rbp-0x4],eax
mov    eax, DWORD PTR [rbp - 0x18]
mov DWORD PTR [rbp-0x8],eax
mov    edx, DWORD PTR [rbp - 8]
mov    eax, DWORD PTR [rbp - 4]
mov esi,edx
mov edi,eax
call BFunc
mov eax,8
leave
ret
```



```
AFunc
push rbp
mov rbp,rsp
sub rsp,0x18
mov DWORD PTR [rbp-0x14],edi
mov DWORD PTR [rbp-0x18],esi
mov DWORD PTR [rbp-0x4],0x3
mov DWORD PTR [rbp-0x8],0x4
mov eax, DWORD PTR [rbp+0x14]
mov DWORD PTR [rbp-0x4],eax
mov    eax, DWORD PTR [rbp - 0x18]
mov DWORD PTR [rbp-0x8],eax
mov    edx, DWORD PTR [rbp - 8]
mov    eax, DWORD PTR [rbp - 4]
mov esi,edx
mov edi,eax
call BFunc
mov eax,8
leave
```

栈的分布	
执行前的rsp ⟶	old_rbp
	ret_addr
rbp ⟶	new_rbp
rsp ⟶	0x0000000500000006

栈底

```
pwndbg> x/dw $rbp - 0x18
0x7ffdf3b28438: 6
pwndbg> x/dw $rbp - 0x14
0x7ffdf3b2843c: 5
```

```
AFunc
push rbp
mov rbp,rsp
sub rsp,0x18
mov DWORD PTR [rbp-0x14],edi
mov DWORD PTR [rbp-0x18],esi
mov DWORD PTR [rbp-0x4],0x3
mov DWORD PTR [rbp-0x8],0x4
mov eax, DWORD PTR [rbp+0x14]
mov DWORD PTR [rbp-0x4],eax
mov    eax, DWORD PTR [rbp - 0x18]
mov DWORD PTR [rbp-0x8],eax
mov    edx, DWORD PTR [rbp - 8]
mov    eax, DWORD PTR [rbp - 4]
mov esi,edx
mov edi,eax
call BFunc
mov eax,8
leave
ret
```

栈的分布	
执行前的rsp ⟶	old_rbp
	ret_addr
rbp ⟶	new_rbp
	0x0000000300000004
rsp ⟶	0x0000000500000006

栈底

```
pwndbg> x/dw $rbp-4
0x7ffdf3b2844c: 3
pwndbg> x/dw $rbp-8
0x7ffdf3b28448: 4
```

AFunc

```
push rbp
mov rbp,rsp
sub rsp,0x18
mov DWORD PTR [rbp-0x14],edi
mov DWORD PTR [rbp-0x18],esi
mov DWORD PTR [rbp-0x4],0x3
mov DWORD PTR [rbp-0x8],0x4
mov eax, DWORD PTR [rbp+0x14]
mov DWORD PTR [rbp-0x4],eax
mov    eax, DWORD PTR [rbp - 0x18]
mov DWORD PTR [rbp-0x8],eax
mov    edx, DWORD PTR [rbp - 8]
mov    eax, DWORD PTR [rbp - 4]
mov esi,edx
mov edi,eax
call BFunc
mov eax,8
leave
ret
```

栈的分布	
执行前的rsp ⟶	old_rbp
	ret_addr
rbp ⟶	new_rbp
	0x0000000300000004
rsp ⟶	0x0000000500000006

栈底

寄存器	
rax	0x5
rdi	0x5
rsi	0x6

```
AFunc
push rbp
mov rbp,rsp
sub rsp,0x18
mov DWORD PTR [rbp-0x14],edi
mov DWORD PTR [rbp-0x18],esi
mov DWORD PTR [rbp-0x4],0x3
mov DWORD PTR [rbp-0x8],0x4
mov eax, DWORD PTR [rbp+0x14]
mov DWORD PTR [rbp-0x4],eax
mov    eax, DWORD PTR [rbp - 0x18]
mov DWORD PTR [rbp-0x8],eax
mov    edx, DWORD PTR [rbp - 8]
mov    eax, DWORD PTR [rbp - 4]
mov esi,edx
mov edi,eax
call BFunc
mov eax,8
leave
ret
```

栈的分布	
执行前的rsp ⟶	old_rbp
	ret_addr
rbp ⟶	new_rbp
	0x0000000500000004
rsp ⟶	0x0000000500000006

栈底

```
pwndbg> x/dw $rbp-4
0x7ffdf3b2844c: 5
```

AFunc
push rbp
mov rbp,rsp
sub rsp,0x18
mov DWORD PTR [rbp-0x14],edi
mov DWORD PTR [rbp-0x18],esi
mov DWORD PTR [rbp-0x4],0x3
mov DWORD PTR [rbp-0x8],0x4
mov eax, DWORD PTR [rbp+0x14]
mov DWORD PTR [rbp-0x4],eax
mov eax, DWORD PTR [rbp - 0x18]
mov DWORD PTR [rbp-0x8],eax
mov edx, DWORD PTR [rbp - 8]
mov eax, DWORD PTR [rbp - 4]
mov esi,edx
mov edi,eax
call BFunc
mov eax,8
leave
ret

栈的分布

执行前的rsp ⟶	old_rbp
	ret_addr
rbp ⟶	new_rbp
	0x0000000500000006
rsp ⟶	0x0000000500000006

栈底

寄存器

```
pwndbg> x/dw $rbp-8
0x7ffdf3b28448: 6
```

AFunc

```
push rbp
mov rbp,rsp
sub rsp,0x18
mov DWORD PTR [rbp-0x14],edi
mov DWORD PTR [rbp-0x18],esi
mov DWORD PTR [rbp-0x4],0x3
mov DWORD PTR [rbp-0x8],0x4
mov eax, DWORD PTR [rbp+0x14]
mov DWORD PTR [rbp-0x4],eax
mov    eax, DWORD PTR [rbp - 0x18]
mov DWORD PTR [rbp-0x8],eax
mov    edx, DWORD PTR [rbp - 8]
mov    eax, DWORD PTR [rbp - 4]
mov esi,edx
mov edi,eax
call BFunc
mov eax,8
leave
```

栈的分布	
执行前的rsp ⟶	old_rbp
	ret_addr
rbp ⟶	new_rbp
	0x0000000500000006
rsp ⟶	0x0000000500000006

栈底

寄存器	
rax	0x5
rdx	0x6
rdi	0x5
rsi	0x6

AFunc

```
push rbp
mov rbp,rsp
sub rsp,0x18
mov DWORD PTR [rbp-0x14],edi
mov DWORD PTR [rbp-0x18],esi
mov DWORD PTR [rbp-0x4],0x3
mov DWORD PTR [rbp-0x8],0x4
mov eax, DWORD PTR [rbp+0x14]
mov DWORD PTR [rbp-0x4],eax
mov    eax, DWORD PTR [rbp - 0x18]
mov DWORD PTR [rbp-0x8],eax
mov    edx, DWORD PTR [rbp - 8]
mov    eax, DWORD PTR [rbp - 4]
mov esi,edx
mov edi,eax
call BFunc
mov eax,8
leave
ret
```

栈的分布	
执行前的rsp ⟶	old_rbp
	ret_addr
rbp ⟶	new_rbp
	0x0000000500000006
rsp ⟶	0x0000000500000006

栈底

寄存器	
rax	0x5
rdx	0x6
rdi	0x5
rsi	0x6

```
AFunc
push rbp
mov rbp,rsp
sub rsp,0x18
mov DWORD PTR [rbp-0x14],edi
mov DWORD PTR [rbp-0x18],esi
mov DWORD PTR [rbp-0x4],0x3
mov DWORD PTR [rbp-0x8],0x4
mov eax, DWORD PTR [rbp+0x14]
mov DWORD PTR [rbp-0x4],eax
mov    eax, DWORD PTR [rbp - 0x18]
mov DWORD PTR [rbp-0x8],eax
mov    edx, DWORD PTR [rbp - 8]
mov    eax, DWORD PTR [rbp - 4]
mov esi,edx
mov edi,eax
call BFunc
      rdi:0x5
      rsi:0x6
mov eax,8
leave
ret
```

栈的分布	
执行前的rsp ⟶	old_rbp
	ret_addr
rbp ⟶	new_rbp
	0x0000000500000006
rsp ⟶	0x0000000500000006

栈底

寄存器	
rax	0x5
rbx	
rcx	
rdx	0x6
rdi	0x5
rsi	0x6

AFunc

```
push rbp
mov rbp,rsp
sub rsp,0x18
mov DWORD PTR [rbp-0x14],edi
mov DWORD PTR [rbp-0x18],esi
mov DWORD PTR [rbp-0x4],0x3
mov DWORD PTR [rbp-0x8],0x4
mov eax, DWORD PTR [rbp+0x14]
mov DWORD PTR [rbp-0x4],eax
mov    eax, DWORD PTR [rbp - 0x18]
mov DWORD PTR [rbp-0x8],eax
mov    edx, DWORD PTR [rbp - 8]
mov    eax, DWORD PTR [rbp - 4]
mov esi,edx
mov edi,eax
call BFunc
mov eax,8
leave
ret
```

栈的分布	
执行前的rsp ⟶	old_rbp
	ret_addr
rbp ⟶	new_rbp
	0x0000000500000006
rsp ⟶	0x0000000500000006

栈底

寄存器	
rax	0x8
rdx	0x6
rdi	0x5
rsi	0x6

```
AFunc
push rbp
mov rbp,rsp
sub rsp,0x18
mov DWORD PTR [rbp-0x14],edi
mov DWORD PTR [rbp-0x18],esi
mov DWORD PTR [rbp-0x4],0x3
mov DWORD PTR [rbp-0x8],0x4
mov eax, DWORD PTR [rbp+0x14]
mov DWORD PTR [rbp-0x4],eax
mov    eax, DWORD PTR [rbp - 0x18]
mov DWORD PTR [rbp-0x8],eax
mov    edx, DWORD PTR [rbp - 8]
mov    eax, DWORD PTR [rbp - 4]
mov esi,edx
mov edi,eax
call BFunc
mov eax,8
leave
ret
```

PS: leave 等效于mov rsp,rbp;pop rbp

AFunc

```
push rbp
mov rbp,rsp
sub rsp,0x18
mov DWORD PTR [rbp-0x14],edi
mov DWORD PTR [rbp-0x18],esi
mov DWORD PTR [rbp-0x4],0x3
mov DWORD PTR [rbp-0x8],0x4
mov eax, DWORD PTR [rbp+0x14]
mov DWORD PTR [rbp-0x4],eax
mov    eax, DWORD PTR [rbp - 0x18]
mov DWORD PTR [rbp-0x8],eax
mov    edx, DWORD PTR [rbp - 8]
mov    eax, DWORD PTR [rbp - 4]
mov esi,edx
mov edi,eax
call BFunc
mov eax,8
leave
ret
```

栈的分布	
rbp,rsp ⟶	old_rbp
	ret_addr
	new_rbp