Stack Frame

```c
unsigned long long int
square_of(unsigned long long int x)
{
    return x * x;
}

unsigned long long int
sum_squared(unsigned long long int a, unsigned long long int b)
{
    unsigned long long int a2;
    unsigned long long int b2;
    a2 = square_of(a);
    b2 = square_of(b);
    return a2 + b2;
}

unsigned long long int
main(void)
{
    unsigned long long int res;
    res = sum_squared(32,128);
    return 0;
}
```

Stack Frame

```c
unsigned long long int
square_of(unsigned long long int x)
{
    return x * x;
}

unsigned long long int
sum_squared(unsigned long long int a, unsigned
{
    unsigned long long int a2;
    unsigned long long int b2;
    a2 = square_of(a);
    b2 = square_of(b);
    return a2 + b2;
}

unsigned long long int
main(void)
{
    unsigned long long int res;
    res = sum_squared(32,128);
    return 0;
}
```

```
00000000004004d6 <square_of>:
  4004d6: 55                      push   %rbp
  4004d7: 48 89 e5                mov    %rsp,%rbp
  4004da: 48 89 7d f8             mov    %rdi,-0x8(%rbp)
  4004de: 48 8b 45 f8             mov    -0x8(%rbp),%rax
  4004e2: 48 0f af 45 f8          imul   -0x8(%rbp),%rax
  4004e7: 5d                      pop    %rbp
  4004e8: c3                      retq

00000000004004e9 <sum_squared>:
  4004e9: 55                      push   %rbp
  4004ea: 48 89 e5                mov    %rsp,%rbp
  4004ed: 48 83 ec 20             sub    $0x20,%rsp
  4004f1: 48 89 7d e8             mov    %rdi,-0x18(%rbp)
  4004f5: 48 89 75 e0             mov    %rsi,-0x20(%rbp)
  4004f9: 48 8b 45 e8             mov    -0x18(%rbp),%rax
  4004fd: 48 89 c7                mov    %rax,%rdi
  400500: e8 d1 ff ff ff          callq  4004d6 <square_of>
  400505: 48 89 45 f0             mov    %rax,-0x10(%rbp)
  400509: 48 8b 45 e0             mov    -0x20(%rbp),%rax
  40050d: 48 89 c7                mov    %rax,%rdi
  400510: e8 c1 ff ff ff          callq  4004d6 <square_of>
  400515: 48 89 45 f8             mov    %rax,-0x8(%rbp)
  400519: 48 8b 55 f0             mov    -0x10(%rbp),%rdx
  40051d: 48 8b 45 f8             mov    -0x8(%rbp),%rax
  400521: 48 01 d0                add    %rdx,%rax
  400524: c9                      leaveq
  400525: c3                      retq

0000000000400526 <main>:
  400526: 55                      push   %rbp
  400527: 48 89 e5                mov    %rsp,%rbp
  40052a: 48 83 ec 10             sub    $0x10,%rsp
  40052e: be 80 00 00 00          mov    $0x80,%esi
  400533: bf 20 00 00 00          mov    $0x20,%edi
  400538: e8 ac ff ff ff          callq  4004e9 <sum_squared>
  40053d: 48 89 45 f8             mov    %rax,-0x8(%rbp)
  400541: b8 00 00 00 00          mov    $0x0,%eax
  400546: c9                      leaveq
  400547: c3                      retq
  400548: 0f 1f 84 00 00 00 00    nopl   0x0(%rax,%rax,1)
  40054f: 00
```
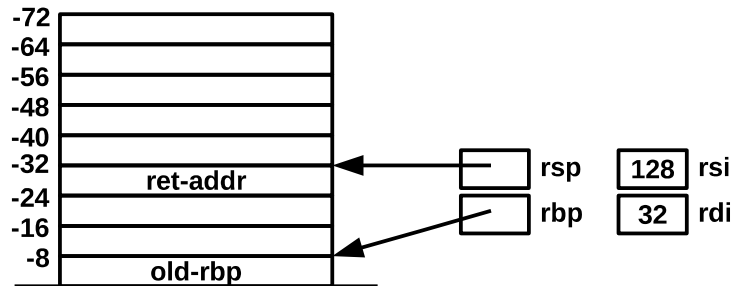
Stack Frame

```c
unsigned long long int
square_of(unsigned long long int x)
{
    return x * x;
}

unsigned long long int
sum_squared(unsigned long long int a, unsigned
{
    unsigned long long int a2;
    unsigned long long int b2;
    a2 = square_of(a);
    b2 = square_of(b);

}

uns
mai
{

}
```



```
00000000004004d6 <square_of>:
  4004d6: 55                    push    %rbp
  4004d7: 48 89 e5              mov     %rsp,%rbp
  4004da: 48 89 7d f8           mov     %rdi,-0x8(%rbp)
  4004de: 48 8b 45 f8           mov     -0x8(%rbp),%rax
  4004e2: 48 0f af 45 f8        imul    -0x8(%rbp),%rax
  4004e7: 5d                    pop     %rbp
  4004e8: c3                    retq

00000000004004e9 <sum_squared>:
  4004e9: 55                    push    %rbp
  4004ea: 48 89 e5              mov     %rsp,%rbp
  4004ed: 48 83 ec 20           sub     $0x20,%rsp
  4004f1: 48 89 7d e8           mov     %rdi,-0x18(%rbp)
  4004f5: 48 89 75 e0           mov     %rsi,-0x20(%rbp)
  4004f9: 48 8b 45 e8           mov     -0x18(%rbp),%rax
  4004fd: 48 89 c7              mov     %rax,%rdi
  400500: e8 d1 ff ff ff        callq   4004d6 <square_of>
  400505: 48 89 45 f0           mov     %rax,-0x10(%rbp)
  400509: 48 8b 45 e0           mov     -0x20(%rbp),%rax
  40050d: 48 89 c7              mov     %rax,%rdi
  400510: e8 c1 ff ff ff        callq   4004d6 <square_of>
  400515: 48 89 45 f8           mov     %rax,-0x8(%rbp)
  400519: 48 8b 55 f0           mov     -0x10(%rbp),%rdx
  40051d: 48 8b 45 f8           mov     -0x8(%rbp),%rax
  400521: 48 01 d0              add     %rdx,%rax
  400524: c9                    leaveq
  400525: c3                    retq

0000000000400526 <main>:
  400526: 55                    push    %rbp
  400527: 48 89 e5              mov     %rsp,%rbp
  40052a: 48 83 ec 10           sub     $0x10,%rsp
  40052e: be 80 00 00 00        mov     $0x80,%esi
  400533: bf 20 00 00 00        mov     $0x20,%edi
  400538: e8 ac ff ff ff        callq   4004e9 <sum_squared>
  40053d: 48 89 45 f8           mov     %rax,-0x8(%rbp)
  400541: b8 00 00 00 00        mov     $0x0,%eax
  400546: c9                    leaveq
  400547: c3                    retq
  400548: 0f 1f 84 00 00 00 00  nopl    0x0(%rax,%rax,1)
  40054f: 00
```
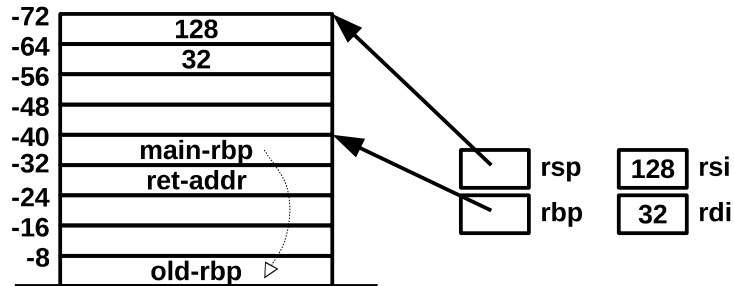
Stack Frame

```c
unsigned long long int
square_of(unsigned long long int x)
{
    return x * x;
}

unsigned long long int
sum_squared(unsigned long long int a, unsigned
{
    unsigned long long int a2;
    unsigned long long int b2;
    a2 = square_of(a);
    b2 = square_of(b);

}

uns
mai
{

}
```

```
00000000004004d6 <square_of>:
  4004d6: 55                   push   %rbp
  4004d7: 48 89 e5             mov    %rsp,%rbp
  4004da: 48 89 7d f8          mov    %rdi,-0x8(%rbp)
  4004de: 48 8b 45 f8          mov    -0x8(%rbp),%rax
  4004e2: 48 0f af 45 f8       imul   -0x8(%rbp),%rax
  4004e7: 5d                   pop    %rbp
  4004e8: c3                   retq

00000000004004e9 <sum_squared>:
  4004e9: 55                   push   %rbp
  4004ea: 48 89 e5             mov    %rsp,%rbp
  4004ed: 48 83 ec 20          sub    $0x20,%rsp
  4004f1: 48 89 7d e8          mov    %rdi,-0x18(%rbp)
  4004f5: 48 89 75 e0          mov    %rsi,-0x20(%rbp)
  4004f9: 48 8b 45 e8          mov    -0x18(%rbp),%rax
  4004fd: 48 89 c7             mov    %rax,%rdi
  400500: e8 d1 ff ff ff       callq  4004d6 <square_of>
  400505: 48 89 45 f0          mov    %rax,-0x10(%rbp)
  400509: 48 8b 45 e0          mov    -0x20(%rbp),%rax
  40050d: 48 89 c7             mov    %rax,%rdi
  400510: e8 c1 ff ff ff       callq  4004d6 <square_of>
  400515: 48 89 45 f8          mov    %rax,-0x8(%rbp)
  400519: 48 8b 55 f0          mov    -0x10(%rbp),%rdx
  40051d: 48 8b 45 f8          mov    -0x8(%rbp),%rax
  400521: 48 01 d0             add    %rdx,%rax
  400524: c9                   leaveq
  400525: c3                   retq

0000000000400526 <main>:
  400526: 55                   push   %rbp
  400527: 48 89 e5             mov    %rsp,%rbp
  40052a: 48 83 ec 10          sub    $0x10,%rsp
  40052e: be 80 00 00 00       mov    $0x80,%esi
  400533: bf 20 00 00 00       mov    $0x20,%edi
  400538: e8 ac ff ff ff       callq  4004e9 <sum_squared>
  40053d: 48 89 45 f8          mov    %rax,-0x8(%rbp)
  400541: b8 00 00 00 00       mov    $0x0,%eax
  400546: c9                   leaveq
  400547: c3                   retq
  400548: 0f 1f 84 00 00 00 00 nopl   0x0(%rax,%rax,1)
  40054f: 00
```

Stack layout:
```
-72   128
-64   32
-56
-48
-40   main-rbp
-32   ret-addr
-24
-16
-8    old-rbp
```

rsp   rbp   128 rsi   32 rdi