

Disassembly of the 'sum' Binary - Function Calls

Tuesday, March 8, 2022 5:48 PM

higher (aka RBP) Z

RBP - 0x8

Start	Stop	Contents
-0x8	-0x0	old RBP contents
-0xc	-0x8	add.f
-0x10	-0xc	add.e
-0x14	-0x10	add.d
-0x18	-0x14	add.c
-0x1c	-0x18	0x4 add.b
-0x20	-0x1c	0x2 add.a
-0x24	-0x20	0x7 total
-0x28	-0x24	0x1 left operand
-0x30	-0x28	
-0x38	-0x30	

lower addresses

lower addr

upper addr

RBP: Z

RSP: Z

EAX: 0x1

left operand

DWORD: 4 bytes

QWORD: 8 bytes

[addr]

"dereference"

*

```

0000000000001139 <main>:
1139: f3 0f 1e fa      endbr64
113d: 55               push    rbp
113e: 48 89 e5         mov     rbp, rsp
1141: 48 83 ec 30      sub     rsp, 0x30
1145: c7 45 d8 01 00 00 00 mov     DWORD PTR [rbp-0x28], 0x1
114c: 48 c7 45 e0 00 00 00 mov     QWORD PTR [rbp-0x20], 0x0
1153: 00
1154: 48 c7 45 e8 00 00 00 mov     QWORD PTR [rbp-0x18], 0x0
115b: 00
115c: 48 c7 45 f0 00 00 00 mov     QWORD PTR [rbp-0x10], 0x0
1163: 00
1164: c7 45 e0 02 00 00 00 mov     DWORD PTR [rbp-0x20], 0x2
116b: c7 45 e4 04 00 00 00 mov     DWORD PTR [rbp-0x1c], 0x4
1172: 8b 45 d8         mov     eax, DWORD PTR [rbp-0x28]
1175: 48 83 ec 08      sub     rsp, 0x8
1179: ff 75 f0         push    QWORD PTR [rbp-0x10]
117c: ff 75 e8         push    QWORD PTR [rbp-0x18]
117f: ff 75 e0         push    QWORD PTR [rbp-0x20]
1182: 89 c7           mov     edi, eax
1184: e8 47 00 00 00  call    11d0 <sum_proxy>
1189: 48 83 c4 20      add     rsp, 0x20
118d: 89 45 dc         mov     DWORD PTR [rbp-0x24], eax
1190: 8b 45 dc         mov     eax, DWORD PTR [rbp-0x24]
1193: 89 c6           mov     esi, eax
1195: 48 8d 3d 68 0e 00 00 lea     rdi, [rip+0xe68] # 2004 <_IO_stdin_used+0x4>
119c: b8 00 00 00 00  mov     eax, 0x0
11a1: e8 8a fe ff ff  call    1030 <printf@plt>
11a6: b8 00 00 00 00  mov     eax, 0x0
11ab: c9             leave   eax, 0x0
11ac: c3             ret
  
```

Semantics: Zero'd out 24 bytes

initializing add.a add.b

left operand

Left operand (!: first argument to sum proxy)

addable:

lower (a) (b) (c) (f) higher

higher

lower

(a) (b) (c) (f)

Start	Stop	Contents
0x38	0x30	??
0x3c	0x38	0 add.f
0x40	0x3c	0 add.e
44	40	0 d
48	44	0 c
4c	48	0x4 b
50	4c	0x2 a

```

0000000000001139 <main>:
1139: f3 0f 1e fa      endbr64
113d: 55               push    rbp
113e: 48 89 e5         mov     rbp, rsp
1141: 48 83 ec 30      sub     rsp, 0x30
1145: c7 45 d8 01 00 00 00 mov     DWORD PTR [rbp-0x28], 0x1
114c: 48 c7 45 e0 00 00 00 mov     QWORD PTR [rbp-0x20], 0x0
1153: 00
1154: 48 c7 45 e8 00 00 00 mov     QWORD PTR [rbp-0x18], 0x0
115b: 00
115c: 48 c7 45 f0 00 00 00 mov     QWORD PTR [rbp-0x10], 0x0
1163: 00
1164: c7 45 e0 02 00 00 00 mov     DWORD PTR [rbp-0x20], 0x2
116b: c7 45 e4 04 00 00 00 mov     DWORD PTR [rbp-0x1c], 0x4
1172: 8b 45 d8         mov     eax, DWORD PTR [rbp-0x28]
1175: 48 83 ec 08      sub     rsp, 0x8
1179: ff 75 f0         push    QWORD PTR [rbp-0x10]
117c: ff 75 e8         push    QWORD PTR [rbp-0x18]
117f: ff 75 e0         push    QWORD PTR [rbp-0x20]
1182: 89 c7           mov     edi, eax
1184: e8 47 00 00 00  call    11d0 <sum_proxy>
1189: 48 83 c4 20      add     rsp, 0x20
118d: 89 45 dc         mov     DWORD PTR [rbp-0x24], eax
1190: 8b 45 dc         mov     eax, DWORD PTR [rbp-0x24]
1193: 89 c6           mov     esi, eax
1195: 48 8d 3d 68 0e 00 00 lea     rdi, [rip+0xe68] # 2004 <_IO_stdin_used+0x4>
119c: b8 00 00 00 00  mov     eax, 0x0
11a1: e8 8a fe ff ff  call    1030 <printf@plt>
11a6: b8 00 00 00 00  mov     eax, 0x0
11ab: c9             leave   eax, 0x0
11ac: c3             ret
  
```