CS342

Operating Systems

Project 3

Muhammad Arham Khan

21701848

Date: 3rd May, 2020

Step 1

Output:

```
pid: 5515
Data:
   uninit1: 6040a0
   uninit2: 603100
   uninit3: 605040
   uninit4: 602160
   init1: 6020a0
   init2: 6020c0
   init3: 6020e0
   init4: 602100
Functions:
   main(): 4007e7
   step1(): 4008ed
   step2(): 400aa1
   step3(): 400b2a
   step4(): 400b4c
   step5(): 400c0f
   step6(): 400d02
```

pmap 5515:

```
5515:   ./app
0000000000400000      8K r-x-- app
0000000000601000      4K r---- app
0000000000602000      4K rw--- app
0000000000603000     12K rw---   [ anon ]
0000000000634000    132K rw---   [ anon ]
00007f3a29328000   1948K r-x-- libc-2.27.so
00007f3a2950f000   2048K ----- libc-2.27.so
00007f3a2970f000     16K r---- libc-2.27.so
00007f3a29713000      8K rw--- libc-2.27.so
00007f3a29715000     16K rw---   [ anon ]
00007f3a29719000    156K r-x-- ld-2.27.so
00007f3a29929000      8K rw---   [ anon ]
00007f3a29940000      4K r---- ld-2.27.so
00007f3a29941000      4K rw--- ld-2.27.so
00007f3a29942000      4K rw---   [ anon ]
00007ffc1cee9000    132K rw---   [ stack ]
00007ffc1cf3c000     12K r----   [ anon ]
00007ffc1cf3f000      4K r-x--   [ anon ]
ffffffffff600000      4K --x--   [ anon ]
```

```
 total         4524K
```

COMMENTS ABOUT PMAP:

From the pmap output it is evident that all of the variables and functions are available in the output and tallying them with the pointer outputs from Step 1, we can see that all the global variables (initialized and uninitialized) and functions are available in the PMAP output. From the output we can see that all the functions have a series like (400…) and hence in the first data set and are executable and also readable. For the global variables' addresses, we can see that all the addresses are in the series like (60…) and hence are present in the third data set and are all executable and readable too.

OBJDUMP –DX APP:

```
app:     file format elf64-x86-64
app
architecture: i386:x86-64, flags 0x00000112:
EXEC_P, HAS_SYMS, D_PAGED
start address 0x0000000000400700

Program Header:
    PHDR off    0x0000000000000040 vaddr 0x0000000000400040 paddr 0x0000000000400040 align 2**3
         filesz 0x00000000000001f8 memsz 0x00000000000001f8 flags r--
  INTERP off    0x0000000000000238 vaddr 0x0000000000400238 paddr 0x0000000000400238 align 2**0
         filesz 0x000000000000001c memsz 0x000000000000001c flags r--
    LOAD off    0x0000000000000000 vaddr 0x0000000000400000 paddr 0x0000000000400000 align 2**21
         filesz 0x0000000000001380 memsz 0x0000000000001380 flags r-x
    LOAD off    0x0000000000001e10 vaddr 0x0000000000601e10 paddr 0x0000000000601e10 align 2**21
         filesz 0x0000000000000270 memsz 0x00000000000041d0 flags rw-
 DYNAMIC off    0x0000000000001e20 vaddr 0x0000000000601e20 paddr 0x0000000000601e20 align 2**3
         filesz 0x00000000000001d0 memsz 0x00000000000001d0 flags rw-
    NOTE off    0x0000000000000254 vaddr 0x0000000000400254 paddr 0x0000000000400254 align 2**2
         filesz 0x0000000000000044 memsz 0x0000000000000044 flags r--
EH_FRAME off    0x00000000000010b0 vaddr 0x00000000004010b0 paddr 0x00000000004010b0 align 2**2
         filesz 0x000000000000008c memsz 0x000000000000008c flags r--
   STACK off    0x0000000000000000 vaddr 0x0000000000000000 paddr 0x0000000000000000 align 2**4
         filesz 0x0000000000000000 memsz 0x0000000000000000 flags rw-
   RELRO off    0x0000000000001e10 vaddr 0x0000000000601e10 paddr 0x0000000000601e10 align 2**0
         filesz 0x00000000000001f0 memsz 0x00000000000001f0 flags r--

Dynamic Section:
  NEEDED               libc.so.6
  INIT                 0x0000000000400620
  FINI                 0x0000000000400e14
  INIT_ARRAY           0x0000000000601e10
  INIT_ARRAYSZ         0x0000000000000008
  FINI_ARRAY           0x0000000000601e18
  FINI_ARRAYSZ         0x0000000000000008
  GNU_HASH             0x0000000000400298
  STRTAB               0x0000000000400408
  SYMTAB               0x00000000004002b8
  STRSZ                0x0000000000000093
  SYMENT               0x0000000000000018
  DEBUG                0x0000000000000000
  PLTGOT               0x0000000000602000
  PLTRELSZ             0x0000000000000108
  PLTREL               0x0000000000000007
  JMPREL               0x0000000000400518
  RELA                 0x00000000004004e8
  RELASZ               0x0000000000000030
  RELAENT              0x0000000000000018
  VERNEED              0x00000000004004b8
  VERNEEDNUM           0x0000000000000001
  VERSYM               0x000000000040049c

Version References:
  required from libc.so.6:
    0x0d696914 0x00 03 GLIBC_2.4
    0x09691a75 0x00 02 GLIBC_2.2.5

Sections:
Idx Name          Size      VMA               LMA               File off  Algn
  0 .interp       0000001c  0000000000400238  0000000000400238  00000238  2**0
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  1 .note.ABI-tag 00000020  0000000000400254  0000000000400254  00000254  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  2 .note.gnu.build-id 00000024  0000000000400274  0000000000400274  00000274  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  3 .gnu.hash     0000001c  0000000000400298  0000000000400298  00000298  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  4 .dynsym       00000150  00000000004002b8  00000000004002b8  000002b8  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  5 .dynstr       00000093  0000000000400408  0000000000400408  00000408  2**0
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  6 .gnu.version  0000001c  000000000040049c  000000000040049c  0000049c  2**1
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  7 .gnu.version_r 00000030  00000000004004b8  00000000004004b8  000004b8  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  8 .rela.dyn     00000030  00000000004004e8  00000000004004e8  000004e8  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  9 .rela.plt     00000108  0000000000400518  0000000000400518  00000518  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
 10 .init         00000017  0000000000400620  0000000000400620  00000620  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
 11 .plt          000000c0  0000000000400640  0000000000400640  00000640  2**4
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
 12 .text         00000712  0000000000400700  0000000000400700  00000700  2**4
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
 13 .fini         00000009  0000000000400e14  0000000000400e14  00000e14  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
 14 .rodata       0000028e  0000000000400e20  0000000000400e20  00000e20  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
 15 .eh_frame_hdr 0000008c  00000000004010b0  00000000004010b0  000010b0  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
 16 .eh_frame     00000240  0000000000401140  0000000000401140  00001140  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
 17 .init_array   00000008  0000000000601e10  0000000000601e10  00001e10  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 18 .fini_array   00000008  0000000000601e18  0000000000601e18  00001e18  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 19 .dynamic      000001d0  0000000000601e20  0000000000601e20  00001e20  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 20 .got          00000010  0000000000601ff0  0000000000601ff0  00001ff0  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 21 .got.plt      00000070  0000000000602000  0000000000602000  00002000  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 22 .data         00000010  0000000000602070  0000000000602070  00002070  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 23 .bss          00003f60  0000000000602080  0000000000602080  00002080  2**5
                  ALLOC
 24 .comment      00000029  0000000000000000  0000000000000000  00002080  2**0
```

```
                CONTENTS, READONLY
SYMBOL TABLE:
0000000000400238 l    d  .interp        0000000000000000              .interp
0000000000400254 l    d  .note.ABI-tag  0000000000000000              .note.ABI-tag
0000000000400274 l    d  .note.gnu.build-id     0000000000000000              .note.gnu.build-id
0000000000400298 l    d  .gnu.hash      0000000000000000              .gnu.hash
00000000004002b8 l    d  .dynsym        0000000000000000              .dynsym
0000000000400408 l    d  .dynstr        0000000000000000              .dynstr
000000000040049c l    d  .gnu.version   0000000000000000              .gnu.version
00000000004004b8 l    d  .gnu.version_r 0000000000000000              .gnu.version_r
00000000004004e8 l    d  .rela.dyn      0000000000000000              .rela.dyn
0000000000400518 l    d  .rela.plt      0000000000000000              .rela.plt
0000000000400620 l    d  .init  0000000000000000              .init
0000000000400640 l    d  .plt   0000000000000000              .plt
0000000000400700 l    d  .text  0000000000000000              .text
0000000000400e14 l    d  .fini  0000000000000000              .fini
0000000000400e20 l    d  .rodata        0000000000000000              .rodata
00000000004010b0 l    d  .eh_frame_hdr  0000000000000000              .eh_frame_hdr
0000000000401140 l    d  .eh_frame      0000000000000000              .eh_frame
0000000000601e10 l    d  .init_array    0000000000000000              .init_array
0000000000601e18 l    d  .fini_array    0000000000000000              .fini_array
0000000000601e20 l    d  .dynamic       0000000000000000              .dynamic
0000000000601ff0 l    d  .got   0000000000000000              .got
0000000000602000 l    d  .got.plt       0000000000000000              .got.plt
0000000000602070 l    d  .data  0000000000000000              .data
0000000000602080 l    d  .bss   0000000000000000              .bss
0000000000000000 l    d  .comment       0000000000000000              .comment
0000000000000000 l    df *ABS*  0000000000000000              crtstuff.c
0000000000400740 l     F .text  0000000000000000              deregister_tm_clones
0000000000400770 l     F .text  0000000000000000              register_tm_clones
00000000004007b0 l     F .text  0000000000000000              __do_global_dtors_aux
0000000000602080 l     O .bss   0000000000000001              completed.7698
0000000000601e18 l     O .fini_array    0000000000000000              __do_global_dtors_aux_fini_array_entry
00000000004007e0 l     F .text  0000000000000000              frame_dummy
0000000000601e10 l     O .init_array    0000000000000000              __frame_dummy_init_array_entry
0000000000000000 l    df *ABS*  0000000000000000              module1.c
0000000000000000 l    df *ABS*  0000000000000000              module2.c
0000000000000000 l    df *ABS*  0000000000000000              module3.c
0000000000000000 l    df *ABS*  0000000000000000              crtstuff.c
000000000040137c l     O .eh_frame      0000000000000000              __FRAME_END__
0000000000000000 l    df *ABS*  0000000000000000
0000000000601e18 l       .init_array    0000000000000000              __init_array_end
0000000000601e20 l     O .dynamic       0000000000000000              _DYNAMIC
0000000000601e10 l       .init_array    0000000000000000              __init_array_start
00000000004010b0 l       .eh_frame_hdr  0000000000000000              __GNU_EH_FRAME_HDR
0000000000602000 l     O .got.plt       0000000000000000              _GLOBAL_OFFSET_TABLE_
0000000000400e10 g     F .text  0000000000000002              __libc_csu_fini
0000000000400c1e g     F .text  00000000000000f3              step5
0000000000400b39 g     F .text  0000000000000022              step3
00000000004008fc g     F .text  00000000000001b4              step1
0000000000602070  w      .data  0000000000000000              data_start
0000000000400d63 g     F .text  0000000000000014              add
0000000000000000       F *UND*  0000000000000000              puts@@GLIBC_2.2.5
0000000000000000       F *UND*  0000000000000000              getpid@@GLIBC_2.2.5
0000000000602080 g       .data  0000000000000000              _edata
0000000000400e14 g     F .fini  0000000000000000              _fini
0000000000400d18 g     F .text  000000000000002c              findAverage
0000000000000000       F *UND*  0000000000000000              __stack_chk_fail@@GLIBC_2.4
0000000000000000       F *UND*  0000000000000000              mmap@@GLIBC_2.2.5
0000000000000000       F *UND*  0000000000000000              printf@@GLIBC_2.2.5
0000000000602140 g     O .bss   0000000000000008              mem_buffer
00000000006020e0 g     O .bss   0000000000000020              init3
00000000006020a0 g     O .bss   0000000000000020              init1
```

```
0000000000000000       F *UND*  0000000000000000            strncat@@GLIBC_2.2.5
0000000000000000       F *UND*  0000000000000000            __libc_start_main@@GLIBC_2.2.5
0000000000602070 g      .data  0000000000000000            __data_start
0000000000000000       F *UND*  0000000000000000            getchar@@GLIBC_2.2.5
0000000000602160 g     O .bss   0000000000000fa0           uninit4
0000000000603100 g     O .bss   0000000000000fa0           uninit2
0000000000000000  w      *UND*  0000000000000000            __gmon_start__
0000000000602078 g     O .data  0000000000000000            .hidden __dso_handle
0000000000602120 g     O .bss   0000000000000004           averageCount
0000000000400e20 g     O .rodata       0000000000000004            _IO_stdin_used
0000000000400b5b g     F .text  00000000000000c3           step4
0000000000400ab0 g     F .text  0000000000000089           step2
0000000000400da0 g     F .text  0000000000000065           __libc_csu_init
0000000000400d11 g     F .text  0000000000000007           step6
0000000000000000       F *UND*  0000000000000000            malloc@@GLIBC_2.2.5
0000000000605fe0 g      .bss   0000000000000000           _end
0000000000400730 g     F .text  0000000000000002           .hidden _dl_relocate_static_pie
0000000000400700 g     F .text  000000000000002b           _start
0000000000602080 g      .bss   0000000000000000           __bss_start
00000000004007e7 g     F .text  0000000000000115           main
0000000000400d77 g     F .text  0000000000000027           recursiveFunction
0000000000400d44 g     F .text  000000000000001f           findSquare
0000000000000000       F *UND*  0000000000000000            open@@GLIBC_2.2.5
0000000000000000       F *UND*  0000000000000000            perror@@GLIBC_2.2.5
0000000000602100 g     O .bss   0000000000000020           init4
00000000006020c0 g     O .bss   0000000000000020           init2
0000000000000000       F *UND*  0000000000000000            exit@@GLIBC_2.2.5
0000000000602080 g     O .data  0000000000000000            .hidden __TMC_END__
00000000006040a0 g     O .bss   0000000000000fa0           uninit1
0000000000605040 g     O .bss   0000000000000fa0           uninit3
0000000000400620 g     F .init  0000000000000000           _init
0000000000602124 g     O .bss   0000000000000004           squareCount



Disassembly of section .init:

0000000000400620 <_init>:
  400620:      48 83 ec 08            sub    $0x8,%rsp
  400624:      48 8b 05 cd 19 20 00    mov    0x2019cd(%rip),%rax        # 601ff8 <__gmon_start__>
  40062b:      48 85 c0              test   %rax,%rax
  40062e:      74 02                 je     400632 <_init+0x12>
  400630:      ff d0                 callq  *%rax
  400632:      48 83 c4 08            add    $0x8,%rsp
  400636:      c3                    retq

Disassembly of section .plt:

0000000000400640 <.plt>:
  400640:      ff 35 c2 19 20 00       pushq  0x2019c2(%rip)        # 602008 <_GLOBAL_OFFSET_TABLE_+0x8>
  400646:      ff 25 c4 19 20 00       jmpq   *0x2019c4(%rip)        # 602010 <_GLOBAL_OFFSET_TABLE_+0x10>
  40064c:      0f 1f 40 00            nopl   0x0(%rax)

0000000000400650 <puts@plt>:
  400650:      ff 25 c2 19 20 00       jmpq   *0x2019c2(%rip)        # 602018 <puts@GLIBC_2.2.5>
  400656:      68 00 00 00 00          pushq  $0x0
  40065b:      e9 e0 ff ff ff         jmpq   400640 <.plt>

0000000000400660 <getpid@plt>:
  400660:      ff 25 ba 19 20 00       jmpq   *0x2019ba(%rip)        # 602020 <getpid@GLIBC_2.2.5>
  400666:      68 01 00 00 00          pushq  $0x1
  40066b:      e9 d0 ff ff ff         jmpq   400640 <.plt>
```

```
0000000000400670 <__stack_chk_fail@plt>:
  400670:     ff 25 b2 19 20 00      jmpq   *0x2019b2(%rip)        # 602028 <__stack_chk_fail@GLIBC_2.4>
  400676:     68 02 00 00 00         pushq  $0x2
  40067b:     e9 c0 ff ff ff         jmpq   400640 <.plt>

0000000000400680 <mmap@plt>:
  400680:     ff 25 aa 19 20 00      jmpq   *0x2019aa(%rip)        # 602030 <mmap@GLIBC_2.2.5>
  400686:     68 03 00 00 00         pushq  $0x3
  40068b:     e9 b0 ff ff ff         jmpq   400640 <.plt>

0000000000400690 <printf@plt>:
  400690:     ff 25 a2 19 20 00      jmpq   *0x2019a2(%rip)        # 602038 <printf@GLIBC_2.2.5>
  400696:     68 04 00 00 00         pushq  $0x4
  40069b:     e9 a0 ff ff ff         jmpq   400640 <.plt>

00000000004006a0 <strncat@plt>:
  4006a0:     ff 25 9a 19 20 00      jmpq   *0x20199a(%rip)        # 602040 <strncat@GLIBC_2.2.5>
  4006a6:     68 05 00 00 00         pushq  $0x5
  4006ab:     e9 90 ff ff ff         jmpq   400640 <.plt>

00000000004006b0 <getchar@plt>:
  4006b0:     ff 25 92 19 20 00      jmpq   *0x201992(%rip)        # 602048 <getchar@GLIBC_2.2.5>
  4006b6:     68 06 00 00 00         pushq  $0x6
  4006bb:     e9 80 ff ff ff         jmpq   400640 <.plt>

00000000004006c0 <malloc@plt>:
  4006c0:     ff 25 8a 19 20 00      jmpq   *0x20198a(%rip)        # 602050 <malloc@GLIBC_2.2.5>
  4006c6:     68 07 00 00 00         pushq  $0x7
  4006cb:     e9 70 ff ff ff         jmpq   400640 <.plt>

00000000004006d0 <open@plt>:
  4006d0:     ff 25 82 19 20 00      jmpq   *0x201982(%rip)        # 602058 <open@GLIBC_2.2.5>
  4006d6:     68 08 00 00 00         pushq  $0x8
  4006db:     e9 60 ff ff ff         jmpq   400640 <.plt>

00000000004006e0 <perror@plt>:
  4006e0:     ff 25 7a 19 20 00      jmpq   *0x20197a(%rip)        # 602060 <perror@GLIBC_2.2.5>
  4006e6:     68 09 00 00 00         pushq  $0x9
  4006eb:     e9 50 ff ff ff         jmpq   400640 <.plt>

00000000004006f0 <exit@plt>:
  4006f0:     ff 25 72 19 20 00      jmpq   *0x201972(%rip)        # 602068 <exit@GLIBC_2.2.5>
  4006f6:     68 0a 00 00 00         pushq  $0xa
  4006fb:     e9 40 ff ff ff         jmpq   400640 <.plt>

Disassembly of section .text:

0000000000400700 <_start>:
  400700:     31 ed                  xor    %ebp,%ebp
  400702:     49 89 d1               mov    %rdx,%r9
  400705:     5e                     pop    %rsi
  400706:     48 89 e2               mov    %rsp,%rdx
  400709:     48 83 e4 f0            and    $0xfffffffffffffff0,%rsp
  40070d:     50                     push   %rax
  40070e:     54                     push   %rsp
  40070f:     49 c7 c0 10 0e 40 00   mov    $0x400e10,%r8
  400716:     48 c7 c1 a0 0d 40 00   mov    $0x400da0,%rcx
  40071d:     48 c7 c7 e7 07 40 00   mov    $0x4007e7,%rdi
  400724:     ff 15 c6 18 20 00      callq  *0x2018c6(%rip)        # 601ff0 <__libc_start_main@GLIBC_2.2.5>
  40072a:     f4                     hlt
  40072b:     0f 1f 44 00 00         nopl   0x0(%rax,%rax,1)

0000000000400730 <_dl_relocate_static_pie>:
  400730:	f3 c3                	repz retq 
  400732:	66 2e 0f 1f 84 00 00 	nopw   %cs:0x0(%rax,%rax,1)
  400739:	00 00 00 
  40073c:	0f 1f 40 00          	nopl   0x0(%rax)

0000000000400740 <deregister_tm_clones>:
  400740:	55                   	push   %rbp
  400741:	b8 80 20 60 00       	mov    $0x602080,%eax
  400746:	48 3d 80 20 60 00    	cmp    $0x602080,%rax
  40074c:	48 89 e5             	mov    %rsp,%rbp
  40074f:	74 17                	je     400768 <deregister_tm_clones+0x28>
  400751:	b8 00 00 00 00       	mov    $0x0,%eax
  400756:	48 85 c0             	test   %rax,%rax
  400759:	74 0d                	je     400768 <deregister_tm_clones+0x28>
  40075b:	5d                   	pop    %rbp
  40075c:	bf 80 20 60 00       	mov    $0x602080,%edi
  400761:	ff e0                	jmpq   *%rax
  400763:	0f 1f 44 00 00       	nopl   0x0(%rax,%rax,1)
  400768:	5d                   	pop    %rbp
  400769:	c3                   	retq   
  40076a:	66 0f 1f 44 00 00    	nopw   0x0(%rax,%rax,1)

0000000000400770 <register_tm_clones>:
  400770:	be 80 20 60 00       	mov    $0x602080,%esi
  400775:	55                   	push   %rbp
  400776:	48 81 ee 80 20 60 00 	sub    $0x602080,%rsi
  40077d:	48 89 e5             	mov    %rsp,%rbp
  400780:	48 c1 fe 03          	sar    $0x3,%rsi
  400784:	48 89 f0             	mov    %rsi,%rax
  400787:	48 c1 e8 3f          	shr    $0x3f,%rax
  40078b:	48 01 c6             	add    %rax,%rsi
  40078e:	48 d1 fe             	sar    %rsi
  400791:	74 15                	je     4007a8 <register_tm_clones+0x38>
  400793:	b8 00 00 00 00       	mov    $0x0,%eax
  400798:	48 85 c0             	test   %rax,%rax
  40079b:	74 0b                	je     4007a8 <register_tm_clones+0x38>
  40079d:	5d                   	pop    %rbp
  40079e:	bf 80 20 60 00       	mov    $0x602080,%edi
  4007a3:	ff e0                	jmpq   *%rax
  4007a5:	0f 1f 00             	nopl   (%rax)
  4007a8:	5d                   	pop    %rbp
  4007a9:	c3                   	retq   
  4007aa:	66 0f 1f 44 00 00    	nopw   0x0(%rax,%rax,1)

00000000004007b0 <__do_global_dtors_aux>:
  4007b0:	80 3d c9 18 20 00 00 	cmpb   $0x0,0x2018c9(%rip)        # 602080 <__TMC_END__>
  4007b7:	75 17                	jne    4007d0 <__do_global_dtors_aux+0x20>
  4007b9:	55                   	push   %rbp
  4007ba:	48 89 e5             	mov    %rsp,%rbp
  4007bd:	e8 7e ff ff ff       	callq  400740 <deregister_tm_clones>
  4007c2:	c6 05 b7 18 20 00 01 	movb   $0x1,0x2018b7(%rip)        # 602080 <__TMC_END__>
  4007c9:	5d                   	pop    %rbp
  4007ca:	c3                   	retq   
  4007cb:	0f 1f 44 00 00       	nopl   0x0(%rax,%rax,1)
  4007d0:	f3 c3                	repz retq 
  4007d2:	0f 1f 40 00          	nopl   0x0(%rax)
  4007d6:	66 2e 0f 1f 84 00 00 	nopw   %cs:0x0(%rax,%rax,1)
  4007dd:	00 00 00 

00000000004007e0 <frame_dummy>:
  4007e0:	55                   	push   %rbp
  4007e1:	48 89 e5             	mov    %rsp,%rbp
  4007e4:	5d                   	pop    %rbp
  4007e5:	eb 89                	jmp    400770 <register_tm_clones>

00000000004007e7 <main>:
  4007e7:	55                   	push   %rbp
  4007e8:	48 89 e5             	mov    %rsp,%rbp
  4007eb:	48 83 ec 20          	sub    $0x20,%rsp
  4007ef:	89 7d ec             	mov    %edi,-0x14(%rbp)
  4007f2:	48 89 75 e0          	mov    %rsi,-0x20(%rbp)
  4007f6:	e8 65 fe ff ff       	callq  400660 <getpid@plt>
  4007fb:	89 45 fc             	mov    %eax,-0x4(%rbp)
  4007fe:	8b 45 fc             	mov    -0x4(%rbp),%eax
  400801:	89 c6                	mov    %eax,%esi
  400803:	48 8d 3d 1e 06 00 00 	lea    0x61e(%rip),%rdi        # 400e28 <_IO_stdin_used+0x8>
  40080a:	b8 00 00 00 00       	mov    $0x0,%eax
  40080f:	e8 7c fe ff ff       	callq  400690 <printf@plt>
  400814:	bf 02 00 00 00       	mov    $0x2,%edi
  400819:	b8 00 00 00 00       	mov    $0x0,%eax
  40081e:	e8 21 05 00 00       	callq  400d44 <findSquare>
  400823:	c6 45 f7 6e          	movb   $0x6e,-0x9(%rbp)
  400827:	c7 45 f8 00 00 00 00 	movl   $0x0,-0x8(%rbp)
  40082e:	e9 b8 00 00 00       	jmpq   4008eb <main+0x104>
  400833:	83 7d f8 05          	cmpl   $0x5,-0x8(%rbp)
  400837:	77 7e                	ja     4008b7 <main+0xd0>
  400839:	8b 45 f8             	mov    -0x8(%rbp),%eax
  40083c:	48 8d 14 85 00 00 00 	lea    0x0(,%rax,4),%rdx
  400843:	00 
  400844:	48 8d 05 09 06 00 00 	lea    0x609(%rip),%rax        # 400e54 <_IO_stdin_used+0x34>
  40084b:	8b 04 02             	mov    (%rdx,%rax,1),%eax
  40084e:	48 63 d0             	movslq %eax,%rdx
  400851:	48 8d 05 fc 05 00 00 	lea    0x5fc(%rip),%rax        # 400e54 <_IO_stdin_used+0x34>
  400858:	48 01 d0             	add    %rdx,%rax
  40085b:	ff e0                	jmpq   *%rax
  40085d:	b8 00 00 00 00       	mov    $0x0,%eax
  400862:	e8 95 00 00 00       	callq  4008fc <step1>
  400867:	83 45 f8 01          	addl   $0x1,-0x8(%rbp)
  40086b:	eb 60                	jmp    4008cd <main+0xe6>
  40086d:	b8 00 00 00 00       	mov    $0x0,%eax
  400872:	e8 39 02 00 00       	callq  400ab0 <step2>
  400877:	83 45 f8 01          	addl   $0x1,-0x8(%rbp)
  40087b:	eb 50                	jmp    4008cd <main+0xe6>
  40087d:	b8 00 00 00 00       	mov    $0x0,%eax
  400882:	e8 b2 02 00 00       	callq  400b39 <step3>
  400887:	83 45 f8 01          	addl   $0x1,-0x8(%rbp)
  40088b:	eb 40                	jmp    4008cd <main+0xe6>
  40088d:	b8 00 00 00 00       	mov    $0x0,%eax
  400892:	e8 c4 02 00 00       	callq  400b5b <step4>
  400897:	83 45 f8 01          	addl   $0x1,-0x8(%rbp)
  40089b:	eb 30                	jmp    4008cd <main+0xe6>
  40089d:	b8 00 00 00 00       	mov    $0x0,%eax
  4008a2:	e8 77 03 00 00       	callq  400c1e <step5>
  4008a7:	83 45 f8 01          	addl   $0x1,-0x8(%rbp)
  4008ab:	eb 20                	jmp    4008cd <main+0xe6>
  4008ad:	b8 00 00 00 00       	mov    $0x0,%eax
  4008b2:	e8 5a 04 00 00       	callq  400d11 <step6>
  4008b7:	48 8d 3d 74 05 00 00 	lea    0x574(%rip),%rdi        # 400e32 <_IO_stdin_used+0x12>
  4008be:	e8 8d fd ff ff       	callq  400650 <puts@plt>
  4008c3:	bf 01 00 00 00       	mov    $0x1,%edi
  4008c8:	e8 23 fe ff ff       	callq  4006f0 <exit@plt>
  4008cd:	48 8d 3d 72 05 00 00 	lea    0x572(%rip),%rdi        # 400e46 <_IO_stdin_used+0x26>
  4008d4:	b8 00 00 00 00       	mov    $0x0,%eax
  4008d9:	e8 b2 fd ff ff       	callq  400690 <printf@plt>
  4008de:	e8 cd fd ff ff       	callq  4006b0 <getchar@plt>
  4008e3:	88 45 f7             	mov    %al,-0x9(%rbp)
  4008e6:	e8 c5 fd ff ff       	callq  4006b0 <getchar@plt>
  4008eb:	80 7d f7 6e          	cmpb   $0x6e,-0x9(%rbp)
  4008ef:	0f 84 3e ff ff ff    	je     400833 <main+0x4c>
  4008f5:	b8 00 00 00 00       	mov    $0x0,%eax
  4008fa:	c9                   	leaveq
  4008fb:	c3                   	retq

00000000004008fc <step1>:
  4008fc:	55                   	push   %rbp
  4008fd:	48 89 e5             	mov    %rsp,%rbp
  400900:	48 8d 3d 65 05 00 00 	lea    0x565(%rip),%rdi        # 400e6c <_IO_stdin_used+0x4c>
  400907:	e8 44 fd ff ff       	callq  400650 <puts@plt>
  40090c:	48 8d 05 8d 37 20 00 	lea    0x20378d(%rip),%rax        # 6040a0 <uninit1>
  400913:	48 89 c6             	mov    %rax,%rsi
  400916:	48 8d 3d 55 05 00 00 	lea    0x555(%rip),%rdi        # 400e72 <_IO_stdin_used+0x52>
  40091d:	b8 00 00 00 00       	mov    $0x0,%eax
  400922:	e8 69 fd ff ff       	callq  400690 <printf@plt>
  400927:	48 8d 05 d2 27 20 00 	lea    0x2027d2(%rip),%rax        # 603100 <uninit2>
  40092e:	48 89 c6             	mov    %rax,%rsi
  400931:	48 8d 3d 4c 05 00 00 	lea    0x54c(%rip),%rdi        # 400e84 <_IO_stdin_used+0x64>
  400938:	b8 00 00 00 00       	mov    $0x0,%eax
  40093d:	e8 4e fd ff ff       	callq  400690 <printf@plt>
  400942:	48 8d 05 f7 46 20 00 	lea    0x2046f7(%rip),%rax        # 605040 <uninit3>
  400949:	48 89 c6             	mov    %rax,%rsi
  40094c:	48 8d 3d 43 05 00 00 	lea    0x543(%rip),%rdi        # 400e96 <_IO_stdin_used+0x76>
  400953:	b8 00 00 00 00       	mov    $0x0,%eax
  400958:	e8 33 fd ff ff       	callq  400690 <printf@plt>
  40095d:	48 8d 05 fc 17 20 00 	lea    0x2017fc(%rip),%rax        # 602160 <uninit4>
  400964:	48 89 c6             	mov    %rax,%rsi
  400967:	48 8d 3d 3a 05 00 00 	lea    0x53a(%rip),%rdi        # 400ea8 <_IO_stdin_used+0x88>
  40096e:	b8 00 00 00 00       	mov    $0x0,%eax
  400973:	e8 18 fd ff ff       	callq  400690 <printf@plt>
  400978:	48 8d 05 21 17 20 00 	lea    0x201721(%rip),%rax        # 6020a0 <init1>
  40097f:	48 89 c6             	mov    %rax,%rsi
  400982:	48 8d 3d 31 05 00 00 	lea    0x531(%rip),%rdi        # 400eba <_IO_stdin_used+0x9a>
  400989:	b8 00 00 00 00       	mov    $0x0,%eax
  40098e:	e8 fd fc ff ff       	callq  400690 <printf@plt>
  400993:	48 8d 05 26 17 20 00 	lea    0x201726(%rip),%rax        # 6020c0 <init2>
  40099a:	48 89 c6             	mov    %rax,%rsi
  40099d:	48 8d 3d 26 05 00 00 	lea    0x526(%rip),%rdi        # 400eca <_IO_stdin_used+0xaa>
  4009a4:	b8 00 00 00 00       	mov    $0x0,%eax
  4009a9:	e8 e2 fc ff ff       	callq  400690 <printf@plt>
  4009ae:	48 8d 05 2b 17 20 00 	lea    0x20172b(%rip),%rax        # 6020e0 <init3>
  4009b5:	48 89 c6             	mov    %rax,%rsi
  4009b8:	48 8d 3d 1b 05 00 00 	lea    0x51b(%rip),%rdi        # 400eda <_IO_stdin_used+0xba>
  4009bf:	b8 00 00 00 00       	mov    $0x0,%eax
  4009c4:	e8 c7 fc ff ff       	callq  400690 <printf@plt>
  4009c9:	48 8d 05 30 17 20 00 	lea    0x201730(%rip),%rax        # 602100 <init4>
  4009d0:	48 89 c6             	mov    %rax,%rsi
  4009d3:	48 8d 3d 10 05 00 00 	lea    0x510(%rip),%rdi        # 400eea <_IO_stdin_used+0xca>
  4009da:	b8 00 00 00 00       	mov    $0x0,%eax
  4009df:	e8 ac fc ff ff       	callq  400690 <printf@plt>
  4009e4:	48 8d 3d 0f 05 00 00 	lea    0x50f(%rip),%rdi        # 400efa <_IO_stdin_used+0xda>
  4009eb:	e8 60 fc ff ff       	callq  400650 <puts@plt>
  4009f0:	48 8d 05 f0 fd ff ff 	lea    -0x210(%rip),%rax        # 4007e7 <main>
  4009f7:	48 89 c6             	mov    %rax,%rsi
  4009fa:	48 8d 3d 04 05 00 00 	lea    0x504(%rip),%rdi        # 400f05 <_IO_stdin_used+0xe5>
  400a01:	b8 00 00 00 00       	mov    $0x0,%eax
  400a06:	e8 85 fc ff ff       	callq  400690 <printf@plt>
  400a0b:	48 8d 05 ea fe ff ff 	lea    -0x116(%rip),%rax        # 4008fc <step1>
  400a12:	48 89 c6             	mov    %rax,%rsi
  400a15:	48 8d 3d fa 04 00 00 	lea    0x4fa(%rip),%rdi        # 400f16 <_IO_stdin_used+0xf6>
  400a1c:	b8 00 00 00 00       	mov    $0x0,%eax
  400a21:	e8 6a fc ff ff       	callq  400690 <printf@plt>
  400a26:	48 8d 05 83 00 00 00 	lea    0x83(%rip),%rax        # 400ab0 <step2>
  400a2d:	48 89 c6             	mov    %rax,%rsi
  400a30:	48 8d 3d f1 04 00 00 	lea    0x4f1(%rip),%rdi        # 400f28 <_IO_stdin_used+0x108>
  400a37:	b8 00 00 00 00       	mov    $0x0,%eax
  400a3c:	e8 4f fc ff ff       	callq  400690 <printf@plt>
  400a41:	48 8d 05 f1 00 00 00 	lea    0xf1(%rip),%rax        # 400b39 <step3>
  400a48:	48 89 c6             	mov    %rax,%rsi
  400a4b:	48 8d 3d e8 04 00 00 	lea    0x4e8(%rip),%rdi        # 400f3a <_IO_stdin_used+0x11a>
  400a52:	b8 00 00 00 00       	mov    $0x0,%eax
  400a57:	e8 34 fc ff ff       	callq  400690 <printf@plt>
  400a5c:	48 8d 05 f8 00 00 00 	lea    0xf8(%rip),%rax        # 400b5b <step4>
  400a63:	48 89 c6             	mov    %rax,%rsi
  400a66:	48 8d 3d df 04 00 00 	lea    0x4df(%rip),%rdi        # 400f4c <_IO_stdin_used+0x12c>
  400a6d:	b8 00 00 00 00       	mov    $0x0,%eax
  400a72:	e8 19 fc ff ff       	callq  400690 <printf@plt>
  400a77:	48 8d 05 a0 01 00 00 	lea    0x1a0(%rip),%rax        # 400c1e <step5>
  400a7e:	48 89 c6             	mov    %rax,%rsi
  400a81:	48 8d 3d d6 04 00 00 	lea    0x4d6(%rip),%rdi        # 400f5e <_IO_stdin_used+0x13e>
  400a88:	b8 00 00 00 00       	mov    $0x0,%eax
  400a8d:	e8 fe fb ff ff       	callq  400690 <printf@plt>
  400a92:	48 8d 05 78 02 00 00 	lea    0x278(%rip),%rax        # 400d11 <step6>
  400a99:	48 89 c6             	mov    %rax,%rsi
  400a9c:	48 8d 3d cd 04 00 00 	lea    0x4cd(%rip),%rdi        # 400f70 <_IO_stdin_used+0x150>
  400aa3:	b8 00 00 00 00       	mov    $0x0,%eax
  400aa8:	e8 e3 fb ff ff       	callq  400690 <printf@plt>
  400aad:	90                   	nop
  400aae:	5d                   	pop    %rbp
  400aaf:	c3                   	retq

0000000000400ab0 <step2>:
  400ab0:	55                   	push   %rbp
  400ab1:	48 89 e5             	mov    %rsp,%rbp
  400ab4:	48 83 ec 20          	sub    $0x20,%rsp
  400ab8:	bf 80 84 1e 00       	mov    $0x1e8480,%edi
  400abd:	e8 fe fb ff ff       	callq  4006c0 <malloc@plt>
  400ac2:	48 89 45 e8          	mov    %rax,-0x18(%rbp)
  400ac6:	bf 40 42 0f 00       	mov    $0xf4240,%edi
  400acb:	e8 f0 fb ff ff       	callq  4006c0 <malloc@plt>
  400ad0:	48 89 45 f0          	mov    %rax,-0x10(%rbp)
  400ad4:	bf 40 42 0f 00       	mov    $0xf4240,%edi
  400ad9:	e8 e2 fb ff ff       	callq  4006c0 <malloc@plt>
  400ade:	48 89 45 f8          	mov    %rax,-0x8(%rbp)
  400ae2:	48 8d 3d 99 04 00 00 	lea    0x499(%rip),%rdi        # 400f82 <_IO_stdin_used+0x162>
  400ae9:	e8 62 fb ff ff       	callq  400650 <puts@plt>
  400aee:	48 8b 45 e8          	mov    -0x18(%rbp),%rax
  400af2:	48 89 c6             	mov    %rax,%rsi
  400af5:	48 8d 3d 98 04 00 00 	lea    0x498(%rip),%rdi        # 400f94 <_IO_stdin_used+0x174>
  400afc:	b8 00 00 00 00       	mov    $0x0,%eax
  400b01:	e8 8a fb ff ff       	callq  400690 <printf@plt>
  400b06:	48 8b 45 f0          	mov    -0x10(%rbp),%rax
  400b0a:	48 89 c6             	mov    %rax,%rsi
  400b0d:	48 8d 3d 94 04 00 00 	lea    0x494(%rip),%rdi        # 400fa8 <_IO_stdin_used+0x188>
  400b14:	b8 00 00 00 00       	mov    $0x0,%eax
  400b19:	e8 72 fb ff ff       	callq  400690 <printf@plt>
  400b1e:	48 8b 45 f8          	mov    -0x8(%rbp),%rax
  400b22:	48 89 c6             	mov    %rax,%rsi
  400b25:	48 8d 3d 90 04 00 00 	lea    0x490(%rip),%rdi        # 400fbc <_IO_stdin_used+0x19c>
  400b2c:	b8 00 00 00 00       	mov    $0x0,%eax
  400b31:	e8 5a fb ff ff       	callq  400690 <printf@plt>
  400b36:	90                   	nop
  400b37:	c9                   	leaveq
  400b38:	c3                   	retq
```

```
0000000000400b39 <step3>:
  400b39:    55                      push   %rbp
  400b3a:    48 89 e5                mov    %rsp,%rbp
  400b3d:    bf 01 00 00 00          mov    $0x1,%edi
  400b42:    b8 00 00 00 00          mov    $0x0,%eax
  400b47:    e8 2b 02 00 00          callq  400d77 <recursiveFunction>
  400b4c:    48 8d 3d 7d 04 00 00    lea    0x47d(%rip),%rdi        # 400fd0 <_IO_stdin_used+0x1b0>
  400b53:    e8 f8 fa ff ff          callq  400650 <puts@plt>
  400b58:    90                      nop
  400b59:    5d                      pop    %rbp
  400b5a:    c3                      retq

0000000000400b5b <step4>:
  400b5b:    55                      push   %rbp
  400b5c:    48 89 e5                mov    %rsp,%rbp
  400b5f:    48 83 ec 10             sub    $0x10,%rsp
  400b63:    ba 80 01 00 00          mov    $0x180,%edx
  400b68:    be 02 00 00 00          mov    $0x2,%esi
  400b6d:    48 8d 3d 78 04 00 00    lea    0x478(%rip),%rdi        # 400fec <_IO_stdin_used+0x1cc>
  400b74:    b8 00 00 00 00          mov    $0x0,%eax
  400b79:    e8 52 fb ff ff          callq  4006d0 <open@plt>
  400b7e:    89 45 f4                mov    %eax,-0xc(%rbp)
  400b81:    83 7d f4 00             cmpl   $0x0,-0xc(%rbp)
  400b85:    79 16                   jns    400b9d <step4+0x42>
  400b87:    48 8d 3d 6a 04 00 00    lea    0x46a(%rip),%rdi        # 400ff8 <_IO_stdin_used+0x1d8>
  400b8e:    e8 4d fb ff ff          callq  4006e0 <perror@plt>
  400b93:    bf 01 00 00 00          mov    $0x1,%edi
  400b98:    e8 53 fb ff ff          callq  4006f0 <exit@plt>
  400b9d:    8b 45 f4                mov    -0xc(%rbp),%eax
  400ba0:    41 b9 00 00 00 00       mov    $0x0,%r9d
  400ba6:    41 89 c0                mov    %eax,%r8d
  400ba9:    b9 02 00 00 00          mov    $0x2,%ecx
  400bae:    ba 03 00 00 00          mov    $0x3,%edx
  400bb3:    be 00 a3 e1 11          mov    $0x11e1a300,%esi
  400bb8:    bf 00 00 00 00          mov    $0x0,%edi
  400bbd:    e8 be fa ff ff          callq  400680 <mmap@plt>
  400bc2:    48 89 45 f8             mov    %rax,-0x8(%rbp)
  400bc6:    48 8b 45 f8             mov    -0x8(%rbp),%rax
  400bca:    48 89 05 6f 15 20 00    mov    %rax,0x20156f(%rip)        # 602140 <mem_buffer>
  400bd1:    48 8b 05 68 15 20 00    mov    0x201568(%rip),%rax        # 602140 <mem_buffer>
  400bd8:    48 83 f8 ff             cmp    $0xffffffffffffffff,%rax
  400bdc:    75 16                   jne    400bf4 <step4+0x99>
  400bde:    48 8d 3d 3a 04 00 00    lea    0x43a(%rip),%rdi        # 40101f <_IO_stdin_used+0x1ff>
  400be5:    e8 f6 fa ff ff          callq  4006e0 <perror@plt>
  400bea:    bf 01 00 00 00          mov    $0x1,%edi
  400bef:    e8 fc fa ff ff          callq  4006f0 <exit@plt>
  400bf4:    48 8d 3d 3e 04 00 00    lea    0x43e(%rip),%rdi        # 401039 <_IO_stdin_used+0x219>
  400bfb:    e8 50 fa ff ff          callq  400650 <puts@plt>
  400c00:    48 8b 05 39 15 20 00    mov    0x201539(%rip),%rax        # 602140 <mem_buffer>
  400c07:    48 89 c6                mov    %rax,%rsi
  400c0a:    48 8d 3d 42 04 00 00    lea    0x442(%rip),%rdi        # 401053 <_IO_stdin_used+0x233>
  400c11:    b8 00 00 00 00          mov    $0x0,%eax
  400c16:    e8 75 fa ff ff          callq  400690 <printf@plt>
  400c1b:    90                      nop
  400c1c:    c9                      leaveq
  400c1d:    c3                      retq

0000000000400c1e <step5>:
  400c1e:    55                      push   %rbp
  400c1f:    48 89 e5                mov    %rsp,%rbp
  400c22:    48 81 ec 70 c3 00 00    sub    $0xc370,%rsp
  400c29:    64 48 8b 04 25 28 00    mov    %fs:0x28,%rax
  400c30:       00 00
  400c32:       48 89 45 f8             mov    %rax,-0x8(%rbp)
  400c36:       31 c0                   xor    %eax,%eax
  400c38:       c7 85 94 3c ff ff 00    movl   $0x0,-0xc36c(%rbp)
  400c3f:       00 00 00
  400c42:       eb 40                   jmp    400c84 <step5+0x66>
  400c44:       48 8b 15 f5 14 20 00    mov    0x2014f5(%rip),%rdx        # 602140 <mem_buffer>
  400c4b:       8b 85 94 3c ff ff       mov    -0xc36c(%rbp),%eax
  400c51:       48 98                   cltq
  400c53:       48 01 d0                add    %rdx,%rax
  400c56:       0f b6 00                movzbl (%rax),%eax
  400c59:       88 85 93 3c ff ff       mov    %al,-0xc36d(%rbp)
  400c5f:       48 8d 8d 93 3c ff ff    lea    -0xc36d(%rbp),%rcx
  400c66:       48 8d 85 a0 3c ff ff    lea    -0xc360(%rbp),%rax
  400c6d:       ba 01 00 00 00          mov    $0x1,%edx
  400c72:       48 89 ce                mov    %rcx,%rsi
  400c75:       48 89 c7                mov    %rax,%rdi
  400c78:       e8 23 fa ff ff          callq  4006a0 <strncat@plt>
  400c7d:       83 85 94 3c ff ff 01    addl   $0x1,-0xc36c(%rbp)
  400c84:       81 bd 94 3c ff ff f3    cmpl   $0x1f3,-0xc36c(%rbp)
  400c8b:       01 00 00
  400c8e:       7e b4                   jle    400c44 <step5+0x26>
  400c90:       48 8d 85 a0 3c ff ff    lea    -0xc360(%rbp),%rax
  400c97:       48 89 c6                mov    %rax,%rsi
  400c9a:       48 8d 3d c8 03 00 00    lea    0x3c8(%rip),%rdi        # 401069 <_IO_stdin_used+0x249>
  400ca1:       b8 00 00 00 00          mov    $0x0,%eax
  400ca6:       e8 e5 f9 ff ff          callq  400690 <printf@plt>
  400cab:       c7 85 98 3c ff ff f4    movl   $0x1f4,-0xc368(%rbp)
  400cb2:       01 00 00
  400cb5:       eb 2b                   jmp    400ce2 <step5+0xc4>
  400cb7:       c7 85 9c 3c ff ff 6f    movl   $0x6f,-0xc364(%rbp)
  400cbe:       00 00 00
  400cc1:       48 8b 15 78 14 20 00    mov    0x201478(%rip),%rdx        # 602140 <mem_buffer>
  400cc8:       8b 85 98 3c ff ff       mov    -0xc368(%rbp),%eax
  400cce:       48 98                   cltq
  400cd0:       48 01 d0                add    %rdx,%rax
  400cd3:       8b 95 9c 3c ff ff       mov    -0xc364(%rbp),%edx
  400cd9:       88 10                   mov    %dl,(%rax)
  400cdb:       83 85 98 3c ff ff 01    addl   $0x1,-0xc368(%rbp)
  400ce2:       81 bd 98 3c ff ff e7    cmpl   $0x3e7,-0xc368(%rbp)
  400ce9:       03 00 00
  400cec:       7e c9                   jle    400cb7 <step5+0x99>
  400cee:       48 8d 3d 8b 03 00 00    lea    0x38b(%rip),%rdi        # 401080 <_IO_stdin_used+0x260>
  400cf5:       e8 56 f9 ff ff          callq  400650 <puts@plt>
  400cfa:       90                      nop
  400cfb:       48 8b 45 f8             mov    -0x8(%rbp),%rax
  400cff:       64 48 33 04 25 28 00    xor    %fs:0x28,%rax
  400d06:       00 00
  400d08:       74 05                   je     400d0f <step5+0xf1>
  400d0a:       e8 61 f9 ff ff          callq  400670 <__stack_chk_fail@plt>
  400d0f:       c9                      leaveq
  400d10:       c3                      retq

0000000000400d11 <step6>:
  400d11:       55                      push   %rbp
  400d12:       48 89 e5                mov    %rsp,%rbp
  400d15:       90                      nop
  400d16:       5d                      pop    %rbp
  400d17:       c3                      retq

0000000000400d18 <findAverage>:
  400d18:       55                      push   %rbp
  400d19:       48 89 e5                mov    %rsp,%rbp
  400d1c:	89 7d fc             	mov    %edi,-0x4(%rbp)
  400d1f:	89 75 f8             	mov    %esi,-0x8(%rbp)
  400d22:	8b 05 f8 13 20 00    	mov    0x2013f8(%rip),%eax        # 602120 <averageCount>
  400d28:	83 c0 01             	add    $0x1,%eax
  400d2b:	89 05 ef 13 20 00    	mov    %eax,0x2013ef(%rip)        # 602120 <averageCount>
  400d31:	8b 55 fc             	mov    -0x4(%rbp),%edx
  400d34:	8b 45 f8             	mov    -0x8(%rbp),%eax
  400d37:	01 d0                	add    %edx,%eax
  400d39:	89 c2                	mov    %eax,%edx
  400d3b:	c1 ea 1f             	shr    $0x1f,%edx
  400d3e:	01 d0                	add    %edx,%eax
  400d40:	d1 f8                	sar    %eax
  400d42:	5d                   	pop    %rbp
  400d43:	c3                   	retq   

0000000000400d44 <findSquare>:
  400d44:	55                   	push   %rbp
  400d45:	48 89 e5             	mov    %rsp,%rbp
  400d48:	89 7d fc             	mov    %edi,-0x4(%rbp)
  400d4b:	8b 05 d3 13 20 00    	mov    0x2013d3(%rip),%eax        # 602124 <squareCount>
  400d51:	83 c0 01             	add    $0x1,%eax
  400d54:	89 05 ca 13 20 00    	mov    %eax,0x2013ca(%rip)        # 602124 <squareCount>
  400d5a:	8b 45 fc             	mov    -0x4(%rbp),%eax
  400d5d:	0f af 45 fc          	imul   -0x4(%rbp),%eax
  400d61:	5d                   	pop    %rbp
  400d62:	c3                   	retq   

0000000000400d63 <add>:
  400d63:	55                   	push   %rbp
  400d64:	48 89 e5             	mov    %rsp,%rbp
  400d67:	89 7d fc             	mov    %edi,-0x4(%rbp)
  400d6a:	89 75 f8             	mov    %esi,-0x8(%rbp)
  400d6d:	8b 55 fc             	mov    -0x4(%rbp),%edx
  400d70:	8b 45 f8             	mov    -0x8(%rbp),%eax
  400d73:	01 d0                	add    %edx,%eax
  400d75:	5d                   	pop    %rbp
  400d76:	c3                   	retq   

0000000000400d77 <recursiveFunction>:
  400d77:	55                   	push   %rbp
  400d78:	48 89 e5             	mov    %rsp,%rbp
  400d7b:	48 83 ec 10          	sub    $0x10,%rsp
  400d7f:	89 7d fc             	mov    %edi,-0x4(%rbp)
  400d82:	81 7d fc 50 46 00 00 	cmpl   $0x4650,-0x4(%rbp)
  400d89:	7f 10                	jg     400d9b <recursiveFunction+0x24>
  400d8b:	83 45 fc 01          	addl   $0x1,-0x4(%rbp)
  400d8f:	8b 45 fc             	mov    -0x4(%rbp),%eax
  400d92:	89 c7                	mov    %eax,%edi
  400d94:	e8 de ff ff ff       	callq  400d77 <recursiveFunction>
  400d99:	eb 01                	jmp    400d9c <recursiveFunction+0x25>
  400d9b:	90                   	nop
  400d9c:	c9                   	leaveq 
  400d9d:	c3                   	retq   
  400d9e:	66 90                	xchg   %ax,%ax

0000000000400da0 <__libc_csu_init>:
  400da0:	41 57                	push   %r15
  400da2:	41 56                	push   %r14
  400da4:	49 89 d7             	mov    %rdx,%r15
  400da7:	41 55                	push   %r13
  400da9:	41 54                	push   %r12
  400dab:	4c 8d 25 5e 10 20 00 	lea    0x20105e(%rip),%r12        # 601e10 <__frame_dummy_init_array_entry>
  400db2:	55                   	push   %rbp
  400db3:	48 8d 2d 5e 10 20 00 	lea    0x20105e(%rip),%rbp        # 601e18 <__init_array_end>
  400dba:	53                   	push   %rbx
  400dbb:	41 89 fd             	mov    %edi,%r13d
  400dbe:	49 89 f6             	mov    %rsi,%r14
  400dc1:	4c 29 e5             	sub    %r12,%rbp
  400dc4:	48 83 ec 08          	sub    $0x8,%rsp
  400dc8:	48 c1 fd 03          	sar    $0x3,%rbp
  400dcc:	e8 4f f8 ff ff       	callq  400620 <_init>
  400dd1:	48 85 ed             	test   %rbp,%rbp
  400dd4:	74 20                	je     400df6 <__libc_csu_init+0x56>
  400dd6:	31 db                	xor    %ebx,%ebx
  400dd8:	0f 1f 84 00 00 00 00 	nopl   0x0(%rax,%rax,1)
  400ddf:	00 
  400de0:	4c 89 fa             	mov    %r15,%rdx
  400de3:	4c 89 f6             	mov    %r14,%rsi
  400de6:	44 89 ef             	mov    %r13d,%edi
  400de9:	41 ff 14 dc          	callq  *(%r12,%rbx,8)
  400ded:	48 83 c3 01          	add    $0x1,%rbx
  400df1:	48 39 dd             	cmp    %rbx,%rbp
  400df4:	75 ea                	jne    400de0 <__libc_csu_init+0x40>
  400df6:	48 83 c4 08          	add    $0x8,%rsp
  400dfa:	5b                   	pop    %rbx
  400dfb:	5d                   	pop    %rbp
  400dfc:	41 5c                	pop    %r12
  400dfe:	41 5d                	pop    %r13
  400e00:	41 5e                	pop    %r14
  400e02:	41 5f                	pop    %r15
  400e04:	c3                   	retq   
  400e05:	90                   	nop
  400e06:	66 2e 0f 1f 84 00 00 	nopw   %cs:0x0(%rax,%rax,1)
  400e0d:	00 00 00 

0000000000400e10 <__libc_csu_fini>:
  400e10:	f3 c3                	repz retq 

Disassembly of section .fini:

0000000000400e14 <_fini>:
  400e14:	48 83 ec 08          	sub    $0x8,%rsp
  400e18:	48 83 c4 08          	add    $0x8,%rsp
  400e1c:	c3                   	retq   
```

COMMENTS ABOUT OBJDUMP:

Comparing some examples, following are the address of the functions/ globals variables in both files:
Step1() function: 4008ed vs 4008ed
init1: 6020a0 vs 6020a0
init2: 6020c0 vs 6020c0
Analyzing the objdump output for the app and comparing it against the pmap, we see that the address are same for functions and global variables in both outputs and hence, can be cross-compared and validated.

OBJDUMP –DX MODULE1.O:

```
module1.o:     file format elf64-x86-64
module1.o
```

```
architecture: i386:x86-64, flags 0x00000011:
HAS_RELOC, HAS_SYMS
start address 0x0000000000000000

Sections:
Idx Name          Size      VMA               LMA               File off  Algn
  0 .text         00000531  0000000000000000  0000000000000000  00000040  2**0
                  CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data         00000000  0000000000000000  0000000000000000  00000571  2**0
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00000080  0000000000000000  0000000000000000  00000580  2**5
                  ALLOC
  3 .rodata       00000286  0000000000000000  0000000000000000  00000580  2**3
                  CONTENTS, ALLOC, LOAD, RELOC, READONLY, DATA
  4 .comment      0000002a  0000000000000000  0000000000000000  00000806  2**0
                  CONTENTS, READONLY
  5 .note.GNU-stack 00000000  0000000000000000  0000000000000000  00000830  2**0
                  CONTENTS, READONLY
  6 .eh_frame     000000f8  0000000000000000  0000000000000000  00000830  2**3
                  CONTENTS, ALLOC, LOAD, RELOC, READONLY, DATA
SYMBOL TABLE:
0000000000000000 l    df *ABS*  0000000000000000 module1.c
0000000000000000 l    d  .text  0000000000000000 .text
0000000000000000 l    d  .data  0000000000000000 .data
0000000000000000 l    d  .bss   0000000000000000 .bss
0000000000000000 l    d  .rodata        0000000000000000 .rodata
0000000000000000 l    d  .note.GNU-stack        0000000000000000 .note.GNU-stack
0000000000000000 l    d  .eh_frame      0000000000000000 .eh_frame
0000000000000000 l    d  .comment       0000000000000000 .comment
0000000000000000 g     O .bss   0000000000000020 init1
0000000000000020 g     O .bss   0000000000000020 init2
0000000000000040 g     O .bss   0000000000000020 init3
0000000000000060 g     O .bss   0000000000000020 init4
0000000000000fa0       O *COM*  0000000000000020 uninit1
0000000000000fa0       O *COM*  0000000000000020 uninit2
0000000000000fa0       O *COM*  0000000000000020 uninit3
0000000000000fa0       O *COM*  0000000000000020 uninit4
0000000000000008       O *COM*  0000000000000008 mem_buffer
0000000000000000 g     F .text  0000000000000115 main
0000000000000000         *UND*  0000000000000000 _GLOBAL_OFFSET_TABLE_
0000000000000000         *UND*  0000000000000000 getpid
0000000000000000         *UND*  0000000000000000 printf
0000000000000000         *UND*  0000000000000000 findSquare
0000000000000115 g     F .text  00000000000001b4 step1
00000000000002c9 g     F .text  0000000000000089 step2
0000000000000352 g     F .text  0000000000000022 step3
0000000000000374 g     F .text  00000000000000c3 step4
0000000000000437 g     F .text  00000000000000f3 step5
000000000000052a g     F .text  0000000000000007 step6
0000000000000000         *UND*  0000000000000000 puts
0000000000000000         *UND*  0000000000000000 exit
```

```
0000000000000000        *UND*  0000000000000000 getchar
0000000000000000        *UND*  0000000000000000 malloc
0000000000000000        *UND*  0000000000000000 recursiveFunction
0000000000000000        *UND*  0000000000000000 open
0000000000000000        *UND*  0000000000000000 perror
0000000000000000        *UND*  0000000000000000 mmap
0000000000000000        *UND*  0000000000000000 strncat
0000000000000000        *UND*  0000000000000000 __stack_chk_fail



Disassembly of section .text:

0000000000000000 <main>:
   0:   55                      push   %rbp
   1:   48 89 e5                mov    %rsp,%rbp
   4:   48 83 ec 20             sub    $0x20,%rsp
   8:   89 7d ec                mov    %edi,-0x14(%rbp)
   b:   48 89 75 e0             mov    %rsi,-0x20(%rbp)
   f:   e8 00 00 00 00          callq  14 <main+0x14>
                        10: R_X86_64_PLT32      getpid-0x4
  14:   89 45 fc                mov    %eax,-0x4(%rbp)
  17:   8b 45 fc                mov    -0x4(%rbp),%eax
  1a:   89 c6                   mov    %eax,%esi
  1c:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 23 <main+0x23>
                        1f: R_X86_64_PC32       .rodata-0x4
  23:   b8 00 00 00 00          mov    $0x0,%eax
  28:   e8 00 00 00 00          callq  2d <main+0x2d>
                        29: R_X86_64_PLT32      printf-0x4
  2d:   bf 02 00 00 00          mov    $0x2,%edi
  32:   b8 00 00 00 00          mov    $0x0,%eax
  37:   e8 00 00 00 00          callq  3c <main+0x3c>
                        38: R_X86_64_PLT32      findSquare-0x4
  3c:   c6 45 f7 6e             movb   $0x6e,-0x9(%rbp)
  40:   c7 45 f8 00 00 00 00    movl   $0x0,-0x8(%rbp)
  47:   e9 b8 00 00 00          jmpq   104 <main+0x104>
  4c:   83 7d f8 05             cmpl   $0x5,-0x8(%rbp)
  50:   77 7e                   ja     d0 <main+0xd0>
  52:   8b 45 f8                mov    -0x8(%rbp),%eax
  55:   48 8d 14 85 00 00 00    lea    0x0(,%rax,4),%rdx
  5c:   00
  5d:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 64 <main+0x64>
                        60: R_X86_64_PC32       .rodata+0x28
  64:   8b 04 02                mov    (%rdx,%rax,1),%eax
  67:   48 63 d0                movslq %eax,%rdx
  6a:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 71 <main+0x71>
                        6d: R_X86_64_PC32       .rodata+0x28
  71:   48 01 d0                add    %rdx,%rax
  74:   ff e0                   jmpq   *%rax
  76:   b8 00 00 00 00          mov    $0x0,%eax
  7b:   e8 00 00 00 00          callq  80 <main+0x80>
```

```
                7c: R_X86_64_PC32      step1-0x4
  80:   83 45 f8 01             addl   $0x1,-0x8(%rbp)
  84:   eb 60                   jmp    e6 <main+0xe6>
  86:   b8 00 00 00 00          mov    $0x0,%eax
  8b:   e8 00 00 00 00          callq  90 <main+0x90>
                8c: R_X86_64_PC32      step2-0x4
  90:   83 45 f8 01             addl   $0x1,-0x8(%rbp)
  94:   eb 50                   jmp    e6 <main+0xe6>
  96:   b8 00 00 00 00          mov    $0x0,%eax
  9b:   e8 00 00 00 00          callq  a0 <main+0xa0>
                9c: R_X86_64_PC32      step3-0x4
  a0:   83 45 f8 01             addl   $0x1,-0x8(%rbp)
  a4:   eb 40                   jmp    e6 <main+0xe6>
  a6:   b8 00 00 00 00          mov    $0x0,%eax
  ab:   e8 00 00 00 00          callq  b0 <main+0xb0>
                ac: R_X86_64_PC32      step4-0x4
  b0:   83 45 f8 01             addl   $0x1,-0x8(%rbp)
  b4:   eb 30                   jmp    e6 <main+0xe6>
  b6:   b8 00 00 00 00          mov    $0x0,%eax
  bb:   e8 00 00 00 00          callq  c0 <main+0xc0>
                bc: R_X86_64_PC32      step5-0x4
  c0:   83 45 f8 01             addl   $0x1,-0x8(%rbp)
  c4:   eb 20                   jmp    e6 <main+0xe6>
  c6:   b8 00 00 00 00          mov    $0x0,%eax
  cb:   e8 00 00 00 00          callq  d0 <main+0xd0>
                cc: R_X86_64_PC32      step6-0x4
  d0:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # d7 <main+0xd7>
                d3: R_X86_64_PC32      .rodata+0x6
  d7:   e8 00 00 00 00          callq  dc <main+0xdc>
                d8: R_X86_64_PLT32     puts-0x4
  dc:   bf 01 00 00 00          mov    $0x1,%edi
  e1:   e8 00 00 00 00          callq  e6 <main+0xe6>
                e2: R_X86_64_PLT32     exit-0x4
  e6:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # ed <main+0xed>
                e9: R_X86_64_PC32      .rodata+0x1a
  ed:   b8 00 00 00 00          mov    $0x0,%eax
  f2:   e8 00 00 00 00          callq  f7 <main+0xf7>
                f3: R_X86_64_PLT32     printf-0x4
  f7:   e8 00 00 00 00          callq  fc <main+0xfc>
                f8: R_X86_64_PLT32     getchar-0x4
  fc:   88 45 f7                mov    %al,-0x9(%rbp)
  ff:   e8 00 00 00 00          callq  104 <main+0x104>
                100: R_X86_64_PLT32    getchar-0x4
 104:   80 7d f7 6e             cmpb   $0x6e,-0x9(%rbp)
 108:   0f 84 3e ff ff ff       je     4c <main+0x4c>
 10e:   b8 00 00 00 00          mov    $0x0,%eax
 113:   c9                      leaveq
 114:   c3                      retq

0000000000000115 <step1>:
 115:   55                      push   %rbp
```

```
116:   48 89 e5                mov    %rsp,%rbp
119:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 120 <step1+0xb>
                 11c: R_X86_64_PC32      .rodata+0x40
120:   e8 00 00 00 00          callq  125 <step1+0x10>
                 121: R_X86_64_PLT32     puts-0x4
125:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 12c <step1+0x17>
                 128: R_X86_64_PC32      uninit1-0x4
12c:   48 89 c6                mov    %rax,%rsi
12f:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 136 <step1+0x21>
                 132: R_X86_64_PC32      .rodata+0x46
136:   b8 00 00 00 00          mov    $0x0,%eax
13b:   e8 00 00 00 00          callq  140 <step1+0x2b>
                 13c: R_X86_64_PLT32     printf-0x4
140:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 147 <step1+0x32>
                 143: R_X86_64_PC32      uninit2-0x4
147:   48 89 c6                mov    %rax,%rsi
14a:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 151 <step1+0x3c>
                 14d: R_X86_64_PC32      .rodata+0x58
151:   b8 00 00 00 00          mov    $0x0,%eax
156:   e8 00 00 00 00          callq  15b <step1+0x46>
                 157: R_X86_64_PLT32     printf-0x4
15b:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 162 <step1+0x4d>
                 15e: R_X86_64_PC32      uninit3-0x4
162:   48 89 c6                mov    %rax,%rsi
165:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 16c <step1+0x57>
                 168: R_X86_64_PC32      .rodata+0x6a
16c:   b8 00 00 00 00          mov    $0x0,%eax
171:   e8 00 00 00 00          callq  176 <step1+0x61>
                 172: R_X86_64_PLT32     printf-0x4
176:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 17d <step1+0x68>
                 179: R_X86_64_PC32      uninit4-0x4
17d:   48 89 c6                mov    %rax,%rsi
180:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 187 <step1+0x72>
                 183: R_X86_64_PC32      .rodata+0x7c
187:   b8 00 00 00 00          mov    $0x0,%eax
18c:   e8 00 00 00 00          callq  191 <step1+0x7c>
                 18d: R_X86_64_PLT32     printf-0x4
191:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 198 <step1+0x83>
                 194: R_X86_64_PC32      init1-0x4
198:   48 89 c6                mov    %rax,%rsi
19b:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 1a2 <step1+0x8d>
                 19e: R_X86_64_PC32      .rodata+0x8e
1a2:   b8 00 00 00 00          mov    $0x0,%eax
1a7:   e8 00 00 00 00          callq  1ac <step1+0x97>
                 1a8: R_X86_64_PLT32     printf-0x4
1ac:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 1b3 <step1+0x9e>
                 1af: R_X86_64_PC32      init2-0x4
1b3:   48 89 c6                mov    %rax,%rsi
1b6:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 1bd <step1+0xa8>
                 1b9: R_X86_64_PC32      .rodata+0x9e
1bd:   b8 00 00 00 00          mov    $0x0,%eax
```

```
1c2:   e8 00 00 00 00          callq  1c7 <step1+0xb2>
                1c3: R_X86_64_PLT32     printf-0x4
1c7:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 1ce <step1+0xb9>
                1ca: R_X86_64_PC32      init3-0x4
1ce:   48 89 c6                mov    %rax,%rsi
1d1:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 1d8 <step1+0xc3>
                1d4: R_X86_64_PC32      .rodata+0xae
1d8:   b8 00 00 00 00          mov    $0x0,%eax
1dd:   e8 00 00 00 00          callq  1e2 <step1+0xcd>
                1de: R_X86_64_PLT32     printf-0x4
1e2:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 1e9 <step1+0xd4>
                1e5: R_X86_64_PC32      init4-0x4
1e9:   48 89 c6                mov    %rax,%rsi
1ec:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 1f3 <step1+0xde>
                1ef: R_X86_64_PC32      .rodata+0xbe
1f3:   b8 00 00 00 00          mov    $0x0,%eax
1f8:   e8 00 00 00 00          callq  1fd <step1+0xe8>
                1f9: R_X86_64_PLT32     printf-0x4
1fd:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 204 <step1+0xef>
                200: R_X86_64_PC32      .rodata+0xce
204:   e8 00 00 00 00          callq  209 <step1+0xf4>
                205: R_X86_64_PLT32     puts-0x4
209:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 210 <step1+0xfb>
                20c: R_X86_64_PC32      main-0x4
210:   48 89 c6                mov    %rax,%rsi
213:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 21a <step1+0x105>
                216: R_X86_64_PC32      .rodata+0xd9
21a:   b8 00 00 00 00          mov    $0x0,%eax
21f:   e8 00 00 00 00          callq  224 <step1+0x10f>
                220: R_X86_64_PLT32     printf-0x4
224:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 22b <step1+0x116>
                227: R_X86_64_PC32      step1-0x4
22b:   48 89 c6                mov    %rax,%rsi
22e:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 235 <step1+0x120>
                231: R_X86_64_PC32      .rodata+0xea
235:   b8 00 00 00 00          mov    $0x0,%eax
23a:   e8 00 00 00 00          callq  23f <step1+0x12a>
                23b: R_X86_64_PLT32     printf-0x4
23f:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 246 <step1+0x131>
                242: R_X86_64_PC32      step2-0x4
246:   48 89 c6                mov    %rax,%rsi
249:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 250 <step1+0x13b>
                24c: R_X86_64_PC32      .rodata+0xfc
250:   b8 00 00 00 00          mov    $0x0,%eax
255:   e8 00 00 00 00          callq  25a <step1+0x145>
                256: R_X86_64_PLT32     printf-0x4
25a:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 261 <step1+0x14c>
                25d: R_X86_64_PC32      step3-0x4
261:   48 89 c6                mov    %rax,%rsi
264:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 26b <step1+0x156>
                267: R_X86_64_PC32      .rodata+0x10e
```

```
 26b:   b8 00 00 00 00          mov    $0x0,%eax
 270:   e8 00 00 00 00          callq  275 <step1+0x160>
                271: R_X86_64_PLT32     printf-0x4
 275:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 27c <step1+0x167>
                278: R_X86_64_PC32      step4-0x4
 27c:   48 89 c6                mov    %rax,%rsi
 27f:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 286 <step1+0x171>
                282: R_X86_64_PC32      .rodata+0x120
 286:   b8 00 00 00 00          mov    $0x0,%eax
 28b:   e8 00 00 00 00          callq  290 <step1+0x17b>
                28c: R_X86_64_PLT32     printf-0x4
 290:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 297 <step1+0x182>
                293: R_X86_64_PC32      step5-0x4
 297:   48 89 c6                mov    %rax,%rsi
 29a:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 2a1 <step1+0x18c>
                29d: R_X86_64_PC32      .rodata+0x132
 2a1:   b8 00 00 00 00          mov    $0x0,%eax
 2a6:   e8 00 00 00 00          callq  2ab <step1+0x196>
                2a7: R_X86_64_PLT32     printf-0x4
 2ab:   48 8d 05 00 00 00 00    lea    0x0(%rip),%rax        # 2b2 <step1+0x19d>
                2ae: R_X86_64_PC32      step6-0x4
 2b2:   48 89 c6                mov    %rax,%rsi
 2b5:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 2bc <step1+0x1a7>
                2b8: R_X86_64_PC32      .rodata+0x144
 2bc:   b8 00 00 00 00          mov    $0x0,%eax
 2c1:   e8 00 00 00 00          callq  2c6 <step1+0x1b1>
                2c2: R_X86_64_PLT32     printf-0x4
 2c6:   90                      nop
 2c7:   5d                      pop    %rbp
 2c8:   c3                      retq

00000000000002c9 <step2>:
 2c9:   55                      push   %rbp
 2ca:   48 89 e5                mov    %rsp,%rbp
 2cd:   48 83 ec 20             sub    $0x20,%rsp
 2d1:   bf 80 84 1e 00          mov    $0x1e8480,%edi
 2d6:   e8 00 00 00 00          callq  2db <step2+0x12>
                2d7: R_X86_64_PLT32     malloc-0x4
 2db:   48 89 45 e8             mov    %rax,-0x18(%rbp)
 2df:   bf 40 42 0f 00          mov    $0xf4240,%edi
 2e4:   e8 00 00 00 00          callq  2e9 <step2+0x20>
                2e5: R_X86_64_PLT32     malloc-0x4
 2e9:   48 89 45 f0             mov    %rax,-0x10(%rbp)
 2ed:   bf 40 42 0f 00          mov    $0xf4240,%edi
 2f2:   e8 00 00 00 00          callq  2f7 <step2+0x2e>
                2f3: R_X86_64_PLT32     malloc-0x4
 2f7:   48 89 45 f8             mov    %rax,-0x8(%rbp)
 2fb:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 302 <step2+0x39>
                2fe: R_X86_64_PC32      .rodata+0x156
 302:   e8 00 00 00 00          callq  307 <step2+0x3e>
                303: R_X86_64_PLT32     puts-0x4
 307:   48 8b 45 e8             mov    -0x18(%rbp),%rax
 30b:   48 89 c6                mov    %rax,%rsi
 30e:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 315 <step2+0x4c>
                        311: R_X86_64_PC32      .rodata+0x168
 315:   b8 00 00 00 00          mov    $0x0,%eax
 31a:   e8 00 00 00 00          callq  31f <step2+0x56>
                        31b: R_X86_64_PLT32     printf-0x4
 31f:   48 8b 45 f0             mov    -0x10(%rbp),%rax
 323:   48 89 c6                mov    %rax,%rsi
 326:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 32d <step2+0x64>
                        329: R_X86_64_PC32      .rodata+0x17c
 32d:   b8 00 00 00 00          mov    $0x0,%eax
 332:   e8 00 00 00 00          callq  337 <step2+0x6e>
                        333: R_X86_64_PLT32     printf-0x4
 337:   48 8b 45 f8             mov    -0x8(%rbp),%rax
 33b:   48 89 c6                mov    %rax,%rsi
 33e:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 345 <step2+0x7c>
                        341: R_X86_64_PC32      .rodata+0x190
 345:   b8 00 00 00 00          mov    $0x0,%eax
 34a:   e8 00 00 00 00          callq  34f <step2+0x86>
                        34b: R_X86_64_PLT32     printf-0x4
 34f:   90                      nop
 350:   c9                      leaveq
 351:   c3                      retq

0000000000000352 <step3>:
 352:   55                      push   %rbp
 353:   48 89 e5                mov    %rsp,%rbp
 356:   bf 01 00 00 00          mov    $0x1,%edi
 35b:   b8 00 00 00 00          mov    $0x0,%eax
 360:   e8 00 00 00 00          callq  365 <step3+0x13>
                        361: R_X86_64_PLT32     recursiveFunction-0x4
 365:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 36c <step3+0x1a>
                        368: R_X86_64_PC32      .rodata+0x1a4
 36c:   e8 00 00 00 00          callq  371 <step3+0x1f>
                        36d: R_X86_64_PLT32     puts-0x4
 371:   90                      nop
 372:   5d                      pop    %rbp
 373:   c3                      retq

0000000000000374 <step4>:
 374:   55                      push   %rbp
 375:   48 89 e5                mov    %rsp,%rbp
 378:   48 83 ec 10             sub    $0x10,%rsp
 37c:   ba 80 01 00 00          mov    $0x180,%edx
 381:   be 02 00 00 00          mov    $0x2,%esi
 386:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 38d <step4+0x19>
                        389: R_X86_64_PC32      .rodata+0x1c0
 38d:   b8 00 00 00 00          mov    $0x0,%eax
 392:   e8 00 00 00 00          callq  397 <step4+0x23>
                        393: R_X86_64_PLT32     open-0x4
 397:   89 45 f4                mov    %eax,-0xc(%rbp)
 39a:   83 7d f4 00             cmpl   $0x0,-0xc(%rbp)
 39e:   79 16                   jns    3b6 <step4+0x42>
 3a0:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 3a7 <step4+0x33>
             3a3: R_X86_64_PC32     .rodata+0x1cc
 3a7:   e8 00 00 00 00          callq  3ac <step4+0x38>
             3a8: R_X86_64_PLT32    perror-0x4
 3ac:   bf 01 00 00 00          mov    $0x1,%edi
 3b1:   e8 00 00 00 00          callq  3b6 <step4+0x42>
             3b2: R_X86_64_PLT32    exit-0x4
 3b6:   8b 45 f4                mov    -0xc(%rbp),%eax
 3b9:   41 b9 00 00 00 00       mov    $0x0,%r9d
 3bf:   41 89 c0                mov    %eax,%r8d
 3c2:   b9 02 00 00 00          mov    $0x2,%ecx
 3c7:   ba 03 00 00 00          mov    $0x3,%edx
 3cc:   be 00 a3 e1 11          mov    $0x11e1a300,%esi
 3d1:   bf 00 00 00 00          mov    $0x0,%edi
 3d6:   e8 00 00 00 00          callq  3db <step4+0x67>
             3d7: R_X86_64_PLT32    mmap-0x4
 3db:   48 89 45 f8             mov    %rax,-0x8(%rbp)
 3df:   48 8b 45 f8             mov    -0x8(%rbp),%rax
 3e3:   48 89 05 00 00 00 00    mov    %rax,0x0(%rip)        # 3ea <step4+0x76>
             3e6: R_X86_64_PC32     mem_buffer-0x4
 3ea:   48 8b 05 00 00 00 00    mov    0x0(%rip),%rax        # 3f1 <step4+0x7d>
             3ed: R_X86_64_PC32     mem_buffer-0x4
 3f1:   48 83 f8 ff             cmp    $0xffffffffffffffff,%rax
 3f5:   75 16                   jne    40d <step4+0x99>
 3f7:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 3fe <step4+0x8a>
             3fa: R_X86_64_PC32     .rodata+0x1f3
 3fe:   e8 00 00 00 00          callq  403 <step4+0x8f>
             3ff: R_X86_64_PLT32    perror-0x4
 403:   bf 01 00 00 00          mov    $0x1,%edi
 408:   e8 00 00 00 00          callq  40d <step4+0x99>
             409: R_X86_64_PLT32    exit-0x4
 40d:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 414 <step4+0xa0>
             410: R_X86_64_PC32     .rodata+0x20d
 414:   e8 00 00 00 00          callq  419 <step4+0xa5>
             415: R_X86_64_PLT32    puts-0x4
 419:   48 8b 05 00 00 00 00    mov    0x0(%rip),%rax        # 420 <step4+0xac>
             41c: R_X86_64_PC32     mem_buffer-0x4
 420:   48 89 c6                mov    %rax,%rsi
 423:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 42a <step4+0xb6>
             426: R_X86_64_PC32     .rodata+0x227
 42a:   b8 00 00 00 00          mov    $0x0,%eax
 42f:   e8 00 00 00 00          callq  434 <step4+0xc0>
             430: R_X86_64_PLT32    printf-0x4
 434:   90                      nop
 435:   c9                      leaveq
 436:   c3                      retq

0000000000000437 <step5>:
```

```
437:   55                    push   %rbp
438:   48 89 e5              mov    %rsp,%rbp
43b:   48 81 ec 70 c3 00 00    sub    $0xc370,%rsp
442:   64 48 8b 04 25 28 00    mov    %fs:0x28,%rax
449:   00 00
44b:   48 89 45 f8           mov    %rax,-0x8(%rbp)
44f:   31 c0                 xor    %eax,%eax
451:   c7 85 94 3c ff ff 00    movl   $0x0,-0xc36c(%rbp)
458:   00 00 00
45b:   eb 40                 jmp    49d <step5+0x66>
45d:   48 8b 15 00 00 00 00    mov    0x0(%rip),%rdx        # 464 <step5+0x2d>
                460: R_X86_64_PC32      mem_buffer-0x4
464:   8b 85 94 3c ff ff       mov    -0xc36c(%rbp),%eax
46a:   48 98                 cltq
46c:   48 01 d0              add    %rdx,%rax
46f:   0f b6 00              movzbl (%rax),%eax
472:   88 85 93 3c ff ff       mov    %al,-0xc36d(%rbp)
478:   48 8d 8d 93 3c ff ff    lea    -0xc36d(%rbp),%rcx
47f:   48 8d 85 a0 3c ff ff    lea    -0xc360(%rbp),%rax
486:   ba 01 00 00 00          mov    $0x1,%edx
48b:   48 89 ce              mov    %rcx,%rsi
48e:   48 89 c7              mov    %rax,%rdi
491:   e8 00 00 00 00          callq  496 <step5+0x5f>
                492: R_X86_64_PLT32     strncat-0x4
496:   83 85 94 3c ff ff 01    addl   $0x1,-0xc36c(%rbp)
49d:   81 bd 94 3c ff ff f3    cmpl   $0x1f3,-0xc36c(%rbp)
4a4:   01 00 00
4a7:   7e b4                 jle    45d <step5+0x26>
4a9:   48 8d 85 a0 3c ff ff    lea    -0xc360(%rbp),%rax
4b0:   48 89 c6              mov    %rax,%rsi
4b3:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 4ba <step5+0x83>
                4b6: R_X86_64_PC32      .rodata+0x23d
4ba:   b8 00 00 00 00          mov    $0x0,%eax
4bf:   e8 00 00 00 00          callq  4c4 <step5+0x8d>
                4c0: R_X86_64_PLT32     printf-0x4
4c4:   c7 85 98 3c ff ff f4    movl   $0x1f4,-0xc368(%rbp)
4cb:   01 00 00
4ce:   eb 2b                 jmp    4fb <step5+0xc4>
4d0:   c7 85 9c 3c ff ff 6f    movl   $0x6f,-0xc364(%rbp)
4d7:   00 00 00
4da:   48 8b 15 00 00 00 00    mov    0x0(%rip),%rdx        # 4e1 <step5+0xaa>
                4dd: R_X86_64_PC32      mem_buffer-0x4
4e1:   8b 85 98 3c ff ff       mov    -0xc368(%rbp),%eax
4e7:   48 98                 cltq
4e9:   48 01 d0              add    %rdx,%rax
4ec:   8b 95 9c 3c ff ff       mov    -0xc364(%rbp),%edx
4f2:   88 10                 mov    %dl,(%rax)
4f4:   83 85 98 3c ff ff 01    addl   $0x1,-0xc368(%rbp)
4fb:   81 bd 98 3c ff ff e7    cmpl   $0x3e7,-0xc368(%rbp)
502:   03 00 00
505:   7e c9                 jle    4d0 <step5+0x99>
```

```
 507:   48 8d 3d 00 00 00 00    lea    0x0(%rip),%rdi        # 50e <step5+0xd7>
                 50a: R_X86_64_PC32      .rodata+0x254
 50e:   e8 00 00 00 00          callq  513 <step5+0xdc>
                 50f: R_X86_64_PLT32     puts-0x4
 513:   90                      nop
 514:   48 8b 45 f8             mov    -0x8(%rbp),%rax
 518:   64 48 33 04 25 28 00    xor    %fs:0x28,%rax
 51f:   00 00
 521:   74 05                   je     528 <step5+0xf1>
 523:   e8 00 00 00 00          callq  528 <step5+0xf1>
                 524: R_X86_64_PLT32     __stack_chk_fail-0x4
 528:   c9                      leaveq
 529:   c3                      retq

000000000000052a <step6>:
 52a:   55                      push   %rbp
 52b:   48 89 e5                mov    %rsp,%rbp
 52e:   90                      nop
 52f:   5d                      pop    %rbp
 530:   c3                      retq
```

OBJDUMP –DX MODULE2.O:

```
module2.o:     file format elf64-x86-64
module2.o
architecture: i386:x86-64, flags 0x00000011:
HAS_RELOC, HAS_SYMS
start address 0x0000000000000000

Sections:
Idx Name          Size      VMA               LMA               File off  Algn
  0 .text         0000004b  0000000000000000  0000000000000000  00000040  2**0
                  CONTENTS, ALLOC, LOAD, RELOC, READONLY, CODE
  1 .data         00000000  0000000000000000  0000000000000000  0000008b  2**0
                  CONTENTS, ALLOC, LOAD, DATA
  2 .bss          00000008  0000000000000000  0000000000000000  0000008c  2**2
                  ALLOC
  3 .comment      0000002a  0000000000000000  0000000000000000  0000008c  2**0
                  CONTENTS, READONLY
  4 .note.GNU-stack 00000000  0000000000000000  0000000000000000  000000b6  2**0
                  CONTENTS, READONLY
  5 .eh_frame     00000058  0000000000000000  0000000000000000  000000b8  2**3
                  CONTENTS, ALLOC, LOAD, RELOC, READONLY, DATA
SYMBOL TABLE:
0000000000000000 l    df *ABS*  0000000000000000 module2.c
0000000000000000 l    d  .text  0000000000000000 .text
0000000000000000 l    d  .data  0000000000000000 .data
0000000000000000 l    d  .bss   0000000000000000 .bss
0000000000000000 l    d  .note.GNU-stack        0000000000000000 .note.GNU-stack
0000000000000000 l    d  .eh_frame      0000000000000000 .eh_frame
```

```
0000000000000000 l    d  .comment      0000000000000000 .comment
0000000000000000 g     O .bss   0000000000000004 averageCount
0000000000000004 g     O .bss   0000000000000004 squareCount
0000000000000000 g     F .text  000000000000002c findAverage
000000000000002c g     F .text  000000000000001f findSquare



Disassembly of section .text:

0000000000000000 <findAverage>:
   0:   55                      push   %rbp
   1:   48 89 e5                mov    %rsp,%rbp
   4:   89 7d fc                mov    %edi,-0x4(%rbp)
   7:   89 75 f8                mov    %esi,-0x8(%rbp)
   a:   8b 05 00 00 00 00       mov    0x0(%rip),%eax        # 10 <findAverage+0x10>
                        c: R_X86_64_PC32        averageCount-0x4
  10:   83 c0 01                add    $0x1,%eax
  13:   89 05 00 00 00 00       mov    %eax,0x0(%rip)        # 19 <findAverage+0x19>
                        15: R_X86_64_PC32       averageCount-0x4
  19:   8b 55 fc                mov    -0x4(%rbp),%edx
  1c:   8b 45 f8                mov    -0x8(%rbp),%eax
  1f:   01 d0                   add    %edx,%eax
  21:   89 c2                   mov    %eax,%edx
  23:   c1 ea 1f                shr    $0x1f,%edx
  26:   01 d0                   add    %edx,%eax
  28:   d1 f8                   sar    %eax
  2a:   5d                      pop    %rbp
  2b:   c3                      retq

000000000000002c <findSquare>:
  2c:   55                      push   %rbp
  2d:   48 89 e5                mov    %rsp,%rbp
  30:   89 7d fc                mov    %edi,-0x4(%rbp)
  33:   8b 05 00 00 00 00       mov    0x0(%rip),%eax        # 39 <findSquare+0xd>
                        35: R_X86_64_PC32       squareCount-0x4
  39:   83 c0 01                add    $0x1,%eax
  3c:   89 05 00 00 00 00       mov    %eax,0x0(%rip)        # 42 <findSquare+0x16>
                        3e: R_X86_64_PC32       squareCount-0x4
  42:   8b 45 fc                mov    -0x4(%rbp),%eax
  45:   0f af 45 fc             imul   -0x4(%rbp),%eax
  49:   5d                      pop    %rbp
  4a:   c3                      retq
```

COMMENTS ABOUT OBJDUMP:

The addressing of the module1.o and module2.o starts from 0x0 and similar to the objdump of the app (first one), I notice that the objdump of module1.o and module2.o are divided into sub-sections like the objdump of app. The subsections are like .text and .data and these same can also be found in objdump of app.

Also, module1.o and module2.o are not linked yet because they are represented as relocatable object modules and so they can also be linked together with other executable objects. Since they are relocatable objects, their addresses also will be starting from 0x0 before linking. Post-linking, the addresses are ordered to avoid any form of overlapping and inconsistency. Similarly, the app objects is also an executable object and so all of its addresses are valid and so they can be executed.

REFERENCES:

As expected, there are two references to the global variable averageCount and these references can be found in objdump of module2.o and objdump of app as well. The instruction mov (move) is used to move a section of memory from memory to the registers and vice versa (between registers). It is also evident that the address is starting with 0x0 as this is a relocatable object (second reference).

```
1.  8b 05 07 14 20 00      mov    0x201407(%rip),%eax       # 602120 <averageCount>
2.  8b 05 00 00 00 00      mov    0x0(%rip),%eax       # 10 <findAverage+0x10>
            c: R_X86_64_PC32       averageCount-0x4
```

Now we present two references made to the function findSquare and the first reference is happening in module1.o's objdump and the second one is in the app's objdump. The function title callq uses and gets the address of the function that is being called.

```
1.  e8 00 00 00 00         callq  3c <main+0x3c>
            38: R_X86_64_PLT32      findSquare-0x4
2.  40081e:      e8 21 05 00 00         callq  400d44 <findSquare>
```

THE STD C LIBRARY :

Here it shows referencing to printf() function of the Standard C library. The first reference happens in objdump of app and the second one happens in module2.o

```
1.  e8 7c fe ff ff         callq  400690 <printf@plt>
2.  e8 00 00 00 00         callq  2d <main+0x2d>
            29: R_X86_64_PLT32      printf-0x4
```

FILE RELATED TO MEMORY MANAGEMENT:

The following files will be related to the memory management and will be kept in the kernel.

- smaps
- smaps_rollup
- status
- limits
- map_files/*
- mem
- numa_maps
- pagemap

STEP 2

OUTPUT:

```
Dynamic pointers:
   pointer 1: 7f6d24447010
   pointer 2: 7f6d23f3a010
   pointer 3: 7f6d23e45010
```

PMAP OF APP:

```
0000000000400000      8K r-x-- app
0000000000601000      4K r---- app
0000000000602000      4K rw--- app
0000000000603000     12K rw---   [ anon ]
000000000111e000    132K rw---   [ anon ]
00007f6d23e45000   1960K rw---   [ anon ]
00007f6d2402f000   1948K r-x-- libc-2.27.so
00007f6d24216000   2048K ----- libc-2.27.so
00007f6d24416000     16K r---- libc-2.27.so
00007f6d2441a000      8K rw--- libc-2.27.so
00007f6d2441c000     16K rw---   [ anon ]
00007f6d24420000    156K r-x-- ld-2.27.so
00007f6d24447000   1964K rw---   [ anon ]
00007f6d24647000      4K r---- ld-2.27.so
00007f6d24648000      4K rw--- ld-2.27.so
00007f6d24649000      4K rw---   [ anon ]
00007ffff3f3e000    132K rw---   [ stack ]
00007ffff3fe8000     12K r----   [ anon ]
00007ffff3feb000      4K r-x--   [ anon ]
ffffffffff600000      4K --x--   [ anon ]
 total             8440K
```

COMMENTS ON PMAP:

The size of the heap (as shown in the pmap output) increases by around 4Mb as we dynamically allocated memory using malloc. I initialized 3 variables using malloc and created objects of 2MB and two objects of around 1 MB. The dynamically allocated variables will be available at 7f6d24447010, 7f6d23f3a010, 7f6d23e45010.

VIRTUAL AND PHYSICAL MEMORY:

The physical memory is 8440KB and the physical memory is 1588KB and this was obtained using pmap –x <process_id>

STEP 3

OUTPUT:

```
 pointer: 7ffcc3c1c81c
   pointer: 7ffcc3c14b1c
   pointer: 7ffcc3c0ce1c
   pointer: 7ffcc3c0511c
   pointer: 7ffcc3bfd41c
   pointer: 7ffcc3bf571c
   pointer: 7ffcc3beda1c
   pointer: 7ffcc3be5d1c
   pointer: 7ffcc3bde01c
   pointer: 7ffcc3bd631c
   pointer: 7ffcc3bce61c
   pointer: 7ffcc3bc691c
   pointer: 7ffcc3bbec1c
   pointer: 7ffcc3bb6f1c
   pointer: 7ffcc3baf21c
   pointer: 7ffcc3ba751c
   pointer: 7ffcc3b9f81c
```

PMAP:

```
0000000000400000      8K r-x-- app
```

```
0000000000601000      4K r---- app
0000000000602000      4K rw--- app
0000000000603000     12K rw---   [ anon ]
00000000017a7000    132K rw---   [ anon ]
00007fb9513da000   1960K rw---   [ anon ]
00007fb9515c4000   1948K r-x-- libc-2.27.so
00007fb9517ab000   2048K ----- libc-2.27.so
00007fb9519ab000     16K r---- libc-2.27.so
00007fb9519af000      8K rw--- libc-2.27.so
00007fb9519b1000     16K rw---   [ anon ]
00007fb9519b5000    156K r-x-- ld-2.27.so
00007fb9519dc000   1964K rw---   [ anon ]
00007fb951bdc000      4K r---- ld-2.27.so
00007fb951bdd000      4K rw--- ld-2.27.so
00007fb951bde000      4K rw---   [ anon ]
00007ffcc3b97000    576K rw---   [ stack ]
00007ffcc3d8d000     12K r----   [ anon ]
00007ffcc3d90000      4K r-x--   [ anon ]
ffffffffff600000      4K  --x--   [ anon ]
 total             8884K
```

COMMENTS ABOUT PMAP:

The pmap shows that the stack size increases from 132KB to 575KB and when we analyse the addresses of the local variable inside the recursive function (val variable), it is evident that all these variables have been declared inside the stack.

STEP 4

OUTPUT:

```
Successfully mapped file.
    map pointer: 7fb93f5bf000
```

PMAP:

```
0000000000400000      8K r-x-- app
0000000000601000      4K r---- app
0000000000602000      4K rw--- app
0000000000603000     12K rw---   [ anon ]
00000000017a7000    132K rw---   [ anon ]
00007fb93f5bf000 292972K rw--- inputfile
00007fb9513da000   1960K rw---   [ anon ]
00007fb9515c4000   1948K r-x-- libc-2.27.so
00007fb9517ab000   2048K ----- libc-2.27.so
00007fb9519ab000     16K r---- libc-2.27.so
00007fb9519af000      8K rw--- libc-2.27.so
00007fb9519b1000     16K rw---   [ anon ]
00007fb9519b5000    156K r-x-- ld-2.27.so
00007fb9519dc000   1964K rw---   [ anon ]
00007fb951bdc000      4K r---- ld-2.27.so
00007fb951bdd000      4K rw--- ld-2.27.so
00007fb951bde000      4K rw---   [ anon ]
00007ffcc3b97000    576K rw---   [ stack ]
00007ffcc3d8d000     12K r----   [ anon ]
00007ffcc3d90000      4K r-x--   [ anon ]
ffffffffff600000      4K --x--   [ anon ]
 total           301856K
```

COMMENTS ABOUT PMAP:

Comparing the output against the Pmap output, we see that the mapped file has physical address: 7fb93f5bf000 and is represented as input file (a specified region) in the pmap file. This segment is the 6th memory segment in the pmap output. Also, since the file size is around 4 MB, the total memory increases from 8884KB to 301856KB as well.

PHYSICAL MEMORY USAGE:

By making use of the pmap –x <process_id> command, we get that the total amount of memory is 301856KB and only 2044KB is in the physical memory (RAM) while the rest is in the virtual memory

STEP 5

PMAP –X of APP:

```
6052:   ./app
Address           Kbytes     RSS   Dirty Mode  Mapping
0000000000400000       8       8       0 r-x-- app
0000000000400000       0       0       0 r-x-- app
0000000000601000       4       4       4 r---- app
0000000000601000       0       0       0 r---- app
0000000000602000       4       4       4 rw--- app
0000000000602000       0       0       0 rw--- app
0000000000603000      12       0       0 rw---   [ anon ]
0000000000603000       0       0       0 rw---   [ anon ]
00000000017a7000     132       4       4 rw---   [ anon ]
00000000017a7000       0       0       0 rw---   [ anon ]
00007fb93f5bf000  292972      64       4 rw--- inputfile
00007fb93f5bf000       0       0       0 rw--- inputfile
00007fb9513da000    1960       8       8 rw---   [ anon ]
00007fb9513da000       0       0       0 rw---   [ anon ]
00007fb9515c4000    1948    1224       0 r-x-- libc-2.27.so
00007fb9515c4000       0       0       0 r-x-- libc-2.27.so
00007fb9517ab000    2048       0       0 ----- libc-2.27.so
00007fb9517ab000       0       0       0 ----- libc-2.27.so
00007fb9519ab000      16      16      16 r---- libc-2.27.so
00007fb9519ab000       0       0       0 r---- libc-2.27.so
00007fb9519af000       8       8       8 rw--- libc-2.27.so
00007fb9519af000       0       0       0 rw--- libc-2.27.so
00007fb9519b1000      16      12      12 rw---   [ anon ]
00007fb9519b1000       0       0       0 rw---   [ anon ]
00007fb9519b5000     156     156       0 r-x-- ld-2.27.so
00007fb9519b5000       0       0       0 r-x-- ld-2.27.so
00007fb9519dc000    1964      12      12 rw---   [ anon ]
00007fb9519dc000       0       0       0 rw---   [ anon ]
00007fb951bdc000       4       4       4 r---- ld-2.27.so
00007fb951bdc000       0       0       0 r---- ld-2.27.so
00007fb951bdd000       4       4       4 rw--- ld-2.27.so
00007fb951bdd000       0       0       0 rw--- ld-2.27.so
00007fb951bde000       4       4       4 rw---   [ anon ]
00007fb951bde000       0       0       0 rw---   [ anon ]
00007ffcc3b97000     576     572     572 rw---   [ stack ]
00007ffcc3b97000       0       0       0 rw---   [ stack ]
00007ffcc3d8d000      12       0       0 r----   [ anon ]
00007ffcc3d8d000       0       0       0 r----   [ anon ]
```

```
00007ffcc3d90000      4      4      0 r-x--   [ anon ]
00007ffcc3d90000      0      0      0 r-x--   [ anon ]
ffffffffff600000      4      0      0 --x--   [ anon ]
ffffffffff600000      0      0      0 --x--   [ anon ]
---------------- ------- ------- -------
total kB         301856    2108    656
```

PMAP:

```
0000000000400000     8K r-x-- app
0000000000601000     4K r---- app
0000000000602000     4K rw--- app
0000000000603000    12K rw---   [ anon ]
00000000017a7000   132K rw---   [ anon ]
00007fb93f5bf000 292972K rw--- inputfile
00007fb9513da000  1960K rw---   [ anon ]
00007fb9515c4000  1948K r-x-- libc-2.27.so
00007fb9517ab000  2048K ----- libc-2.27.so
00007fb9519ab000    16K r---- libc-2.27.so
00007fb9519af000     8K rw--- libc-2.27.so
00007fb9519b1000    16K rw---   [ anon ]
00007fb9519b5000   156K r-x-- ld-2.27.so
00007fb9519dc000  1964K rw---   [ anon ]
00007fb951bdc000     4K r---- ld-2.27.so
00007fb951bdd000     4K rw--- ld-2.27.so
00007fb951bde000     4K rw---   [ anon ]
00007ffcc3b97000   576K rw---   [ stack ]
00007ffcc3d8d000    12K r----   [ anon ]
00007ffcc3d90000     4K r-x--   [ anon ]
ffffffffff600000     4K --x--   [ anon ]
 total          301856K
```

COMMENTS:

In this part, we read around 500 bytes from the mapped file and stored in a char array and hence, this change is visible in the physical memory size usage in the pmap (which has increased from 2044KB to 2108KB. But as expected, the size of the mmap is the same as no change has been made to the already mapped file and nothing new has been allocated in the virtual memory

STEP 6

OUTPUT:

(truncated in terminal)

```
pointer: 400845  :e0 e8 65 fe ff ff 89 45
pointer: 400846  :e8 65 fe ff ff 89 45 fc
pointer: 400847  :65 fe ff ff 89 45 fc 8b
pointer: 400848  :fe ff ff 89 45 fc 8b 45
pointer: 400849  :ff ff 89 45 fc 8b 45 fc
pointer: 40084a  :ff 89 45 fc 8b 45 fc 89
pointer: 40084b  :89 45 fc 8b 45 fc 89 c6
pointer: 40084c  :45 fc 8b 45 fc 89 c6 48
pointer: 40084d  :fc 8b 45 fc 89 c6 48 8d
pointer: 40084e  :8b 45 fc 89 c6 48 8d 3d
pointer: 40084f  :45 fc 89 c6 48 8d 3d ee
pointer: 400850  :fc 89 c6 48 8d 3d ee 6
pointer: 400851  :89 c6 48 8d 3d ee 6 0
pointer: 400852  :c6 48 8d 3d ee 6 0 0
pointer: 400853  :48 8d 3d ee 6 0 0 b8
pointer: 400854  :8d 3d ee 6 0 0 b8 0
pointer: 400855  :3d ee 6 0 0 b8 0 0
pointer: 400856  :ee 6 0 0 b8 0 0 0
pointer: 400857  :6 0 0 b8 0 0 0 0
pointer: 400858  :0 0 b8 0 0 0 0 e8
pointer: 400859  :0 b8 0 0 0 0 e8 7c
pointer: 40085a  :b8 0 0 0 0 e8 7c fe
pointer: 40085b  :0 0 0 0 e8 7c fe ff
pointer: 40085c  :0 0 0 e8 7c fe ff ff
pointer: 40085d  :0 0 e8 7c fe ff ff bf
pointer: 40085e  :0 e8 7c fe ff ff bf 2
pointer: 40085f  :e8 7c fe ff ff bf 2 0
pointer: 400860  :7c fe ff ff bf 2 0 0
pointer: 400861  :fe ff ff bf 2 0 0 0
pointer: 400862  :ff ff bf 2 0 0 0 b8
pointer: 400863  :ff bf 2 0 0 0 b8 0
```

```
pointer: 400864  :bf 2 0 0 0 b8 0 0
pointer: 400865  :2 0 0 0 b8 0 0 0
pointer: 400866  :0 0 0 b8 0 0 0 0
pointer: 400867  :0 0 b8 0 0 0 0 e8
pointer: 400868  :0 b8 0 0 0 0 e8 af
pointer: 400869  :b8 0 0 0 0 e8 af 5
pointer: 40086a  :0 0 0 0 e8 af 5 0
pointer: 40086b  :0 0 0 e8 af 5 0 0
pointer: 40086c  :0 0 e8 af 5 0 0 c6
pointer: 40086d  :0 e8 af 5 0 0 c6 45
pointer: 40086e  :e8 af 5 0 0 c6 45 f7
pointer: 40086f  :af 5 0 0 c6 45 f7 6e
pointer: 400870  :5 0 0 c6 45 f7 6e c7
pointer: 400871  :0 0 c6 45 f7 6e c7 45
pointer: 400872  :0 c6 45 f7 6e c7 45 f8
pointer: 400873  :c6 45 f7 6e c7 45 f8 0
pointer: 400874  :45 f7 6e c7 45 f8 0 0
pointer: 400875  :f7 6e c7 45 f8 0 0 0
pointer: 400876  :6e c7 45 f8 0 0 0 0
pointer: 400877  :c7 45 f8 0 0 0 0 e9
pointer: 400878  :45 f8 0 0 0 0 e9 b8
pointer: 400879  :f8 0 0 0 0 e9 b8 0
pointer: 40087a  :0 0 0 0 e9 b8 0 0
pointer: 40087b  :0 0 0 e9 b8 0 0 0
pointer: 40087c  :0 0 e9 b8 0 0 0 83
pointer: 40087d  :0 e9 b8 0 0 0 83 7d
pointer: 40087e  :e9 b8 0 0 0 83 7d f8
pointer: 40087f  :b8 0 0 0 83 7d f8 5
pointer: 400880  :0 0 0 83 7d f8 5 77
pointer: 400881  :0 0 83 7d f8 5 77 7e
pointer: 400882  :0 83 7d f8 5 77 7e 8b
pointer: 400883  :83 7d f8 5 77 7e 8b 45
pointer: 400884  :7d f8 5 77 7e 8b 45 f8
pointer: 400885  :f8 5 77 7e 8b 45 f8 48
```

```
pointer: 400886  :5 77 7e 8b 45 f8 48 8d
pointer: 400887  :77 7e 8b 45 f8 48 8d 14
pointer: 400888  :7e 8b 45 f8 48 8d 14 85
pointer: 400889  :8b 45 f8 48 8d 14 85 0
pointer: 40088a  :45 f8 48 8d 14 85 0 0
pointer: 40088b  :f8 48 8d 14 85 0 0 0
pointer: 40088c  :48 8d 14 85 0 0 0 0
pointer: 40088d  :8d 14 85 0 0 0 0 48
pointer: 40088e  :14 85 0 0 0 0 48 8d
pointer: 40088f  :85 0 0 0 0 48 8d 5
pointer: 400890  :0 0 0 0 48 8d 5 d9
pointer: 400891  :0 0 0 48 8d 5 d9 6
pointer: 400892  :0 0 48 8d 5 d9 6 0
pointer: 400893  :0 48 8d 5 d9 6 0 0
pointer: 400894  :48 8d 5 d9 6 0 0 8b
pointer: 400895  :8d 5 d9 6 0 0 8b 4
pointer: 400896  :5 d9 6 0 0 8b 4 2
pointer: 400897  :d9 6 0 0 8b 4 2 48
pointer: 400898  :6 0 0 8b 4 2 48 63
pointer: 400899  :0 0 8b 4 2 48 63 d0
pointer: 40089a  :0 8b 4 2 48 63 d0 48
pointer: 40089b  :8b 4 2 48 63 d0 48 8d
pointer: 40089c  :4 2 48 63 d0 48 8d 5
pointer: 40089d  :2 48 63 d0 48 8d 5 cc
pointer: 40089e  :48 63 d0 48 8d 5 cc 6
pointer: 40089f  :63 d0 48 8d 5 cc 6 0
pointer: 4008a0  :d0 48 8d 5 cc 6 0 0
pointer: 4008a1  :48 8d 5 cc 6 0 0 48
pointer: 4008a2  :8d 5 cc 6 0 0 48 1
pointer: 4008a3  :5 cc 6 0 0 48 1 d0
pointer: 4008a4  :cc 6 0 0 48 1 d0 ff
pointer: 4008a5  :6 0 0 48 1 d0 ff e0
pointer: 4008a6  :0 0 48 1 d0 ff e0 b8
pointer: 4008a7  :0 48 1 d0 ff e0 b8 0
```

```
pointer: 4008a8  :48 1 d0 ff e0 b8 0 0
pointer: 4008a9  :1 d0 ff e0 b8 0 0 0
pointer: 4008aa  :d0 ff e0 b8 0 0 0 0
pointer: 4008ab  :ff e0 b8 0 0 0 0 e8
pointer: 4008ac  :e0 b8 0 0 0 0 e8 95
pointer: 4008ad  :b8 0 0 0 0 e8 95 0
pointer: 4008ae  :0 0 0 0 e8 95 0 0
pointer: 4008af  :0 0 0 e8 95 0 0 0
pointer: 4008b0  :0 0 e8 95 0 0 0 83
pointer: 4008b1  :0 e8 95 0 0 0 83 45
pointer: 4008b2  :e8 95 0 0 0 83 45 f8
pointer: 4008b3  :95 0 0 0 83 45 f8 1
pointer: 4008b4  :0 0 0 83 45 f8 1 eb
pointer: 4008b5  :0 0 83 45 f8 1 eb 60
pointer: 4008b6  :0 83 45 f8 1 eb 60 b8
pointer: 4008b7  :83 45 f8 1 eb 60 b8 0
pointer: 4008b8  :45 f8 1 eb 60 b8 0 0
pointer: 4008b9  :f8 1 eb 60 b8 0 0 0
pointer: 4008ba  :1 eb 60 b8 0 0 0 0
pointer: 4008bb  :eb 60 b8 0 0 0 0 e8
pointer: 4008bc  :60 b8 0 0 0 0 e8 39
pointer: 4008bd  :b8 0 0 0 0 e8 39 2
pointer: 4008be  :0 0 0 0 e8 39 2 0
pointer: 4008bf  :0 0 0 e8 39 2 0 0
pointer: 4008c0  :0 0 e8 39 2 0 0 83
pointer: 4008c1  :0 e8 39 2 0 0 83 45
pointer: 4008c2  :e8 39 2 0 0 83 45 f8
pointer: 4008c3  :39 2 0 0 83 45 f8 1
pointer: 4008c4  :2 0 0 83 45 f8 1 eb
pointer: 4008c5  :0 0 83 45 f8 1 eb 50
pointer: 4008c6  :0 83 45 f8 1 eb 50 b8
pointer: 4008c7  :83 45 f8 1 eb 50 b8 0
pointer: 4008c8  :45 f8 1 eb 50 b8 0 0
pointer: 4008c9  :f8 1 eb 50 b8 0 0 0
```

```
pointer: 4008ca  :1 eb 50 b8 0 0 0 0
pointer: 4008cb  :eb 50 b8 0 0 0 0 e8
pointer: 4008cc  :50 b8 0 0 0 0 e8 b2
pointer: 4008cd  :b8 0 0 0 0 e8 b2 2
pointer: 4008ce  :0 0 0 0 e8 b2 2 0
pointer: 4008cf  :0 0 0 e8 b2 2 0 0
pointer: 4008d0  :0 0 e8 b2 2 0 0 83
pointer: 4008d1  :0 e8 b2 2 0 0 83 45
pointer: 4008d2  :e8 b2 2 0 0 83 45 f8
pointer: 4008d3  :b2 2 0 0 83 45 f8 1
pointer: 4008d4  :2 0 0 83 45 f8 1 eb
pointer: 4008d5  :0 0 83 45 f8 1 eb 40
pointer: 4008d6  :0 83 45 f8 1 eb 40 b8
pointer: 4008d7  :83 45 f8 1 eb 40 b8 0
pointer: 4008d8  :45 f8 1 eb 40 b8 0 0
pointer: 4008d9  :f8 1 eb 40 b8 0 0 0
pointer: 4008da  :1 eb 40 b8 0 0 0 0
pointer: 4008db  :eb 40 b8 0 0 0 0 e8
pointer: 4008dc  :40 b8 0 0 0 0 e8 c4
pointer: 4008dd  :b8 0 0 0 0 e8 c4 2
pointer: 4008de  :0 0 0 0 e8 c4 2 0
pointer: 4008df  :0 0 0 e8 c4 2 0 0
pointer: 4008e0  :0 0 e8 c4 2 0 0 83
pointer: 4008e1  :0 e8 c4 2 0 0 83 45
pointer: 4008e2  :e8 c4 2 0 0 83 45 f8
pointer: 4008e3  :c4 2 0 0 83 45 f8 1
pointer: 4008e4  :2 0 0 83 45 f8 1 eb
pointer: 4008e5  :0 0 83 45 f8 1 eb 30
pointer: 4008e6  :0 83 45 f8 1 eb 30 b8
pointer: 4008e7  :83 45 f8 1 eb 30 b8 0
pointer: 4008e8  :45 f8 1 eb 30 b8 0 0
pointer: 4008e9  :f8 1 eb 30 b8 0 0 0
pointer: 4008ea  :1 eb 30 b8 0 0 0 0
pointer: 4008eb  :eb 30 b8 0 0 0 0 e8
```

```
pointer: 4008ec  :30 b8 0 0 0 0 e8 77
pointer: 4008ed  :b8 0 0 0 0 e8 77 3
pointer: 4008ee  :0 0 0 0 e8 77 3 0
pointer: 4008ef  :0 0 0 e8 77 3 0 0
pointer: 4008f0  :0 0 e8 77 3 0 0 83
pointer: 4008f1  :0 e8 77 3 0 0 83 45
pointer: 4008f2  :e8 77 3 0 0 83 45 f8
pointer: 4008f3  :77 3 0 0 83 45 f8 1
pointer: 4008f4  :3 0 0 83 45 f8 1 eb
pointer: 4008f5  :0 0 83 45 f8 1 eb 20
pointer: 4008f6  :0 83 45 f8 1 eb 20 b8
pointer: 4008f7  :83 45 f8 1 eb 20 b8 0
pointer: 4008f8  :45 f8 1 eb 20 b8 0 0
pointer: 4008f9  :f8 1 eb 20 b8 0 0 0
pointer: 4008fa  :1 eb 20 b8 0 0 0 0
pointer: 4008fb  :eb 20 b8 0 0 0 0 e8
pointer: 4008fc  :20 b8 0 0 0 0 e8 5a
pointer: 4008fd  :b8 0 0 0 0 e8 5a 4
pointer: 4008fe  :0 0 0 0 e8 5a 4 0
pointer: 4008ff  :0 0 0 e8 5a 4 0 0
pointer: 400900  :0 0 e8 5a 4 0 0 48
pointer: 400901  :0 e8 5a 4 0 0 48 8d
pointer: 400902  :e8 5a 4 0 0 48 8d 3d
pointer: 400903  :5a 4 0 0 48 8d 3d 44
pointer: 400904  :4 0 0 48 8d 3d 44 6
pointer: 400905  :0 0 48 8d 3d 44 6 0
pointer: 400906  :0 48 8d 3d 44 6 0 0
pointer: 400907  :48 8d 3d 44 6 0 0 e8
pointer: 400908  :8d 3d 44 6 0 0 e8 8d
pointer: 400909  :3d 44 6 0 0 e8 8d fd
pointer: 40090a  :44 6 0 0 e8 8d fd ff
pointer: 40090b  :6 0 0 e8 8d fd ff ff
pointer: 40090c  :0 0 e8 8d fd ff ff bf
pointer: 40090d  :0 e8 8d fd ff ff bf 1
```

```
pointer: 40090e  :e8 8d fd ff ff bf 1 0
pointer: 40090f  :8d fd ff ff bf 1 0 0
pointer: 400910  :fd ff ff bf 1 0 0 0
pointer: 400911  :ff ff bf 1 0 0 0 e8
pointer: 400912  :ff bf 1 0 0 0 e8 23
pointer: 400913  :bf 1 0 0 0 e8 23 fe
pointer: 400914  :1 0 0 0 e8 23 fe ff
pointer: 400915  :0 0 0 e8 23 fe ff ff
pointer: 400916  :0 0 e8 23 fe ff ff 48
pointer: 400917  :0 e8 23 fe ff ff 48 8d
pointer: 400918  :e8 23 fe ff ff 48 8d 3d
pointer: 400919  :23 fe ff ff 48 8d 3d 42
pointer: 40091a  :fe ff ff 48 8d 3d 42 6
pointer: 40091b  :ff ff 48 8d 3d 42 6 0
pointer: 40091c  :ff 48 8d 3d 42 6 0 0
pointer: 40091d  :48 8d 3d 42 6 0 0 b8
pointer: 40091e  :8d 3d 42 6 0 0 b8 0
pointer: 40091f  :3d 42 6 0 0 b8 0 0
pointer: 400920  :42 6 0 0 b8 0 0 0
pointer: 400921  :6 0 0 b8 0 0 0 0
pointer: 400922  :0 0 b8 0 0 0 0 e8
pointer: 400923  :0 b8 0 0 0 0 e8 b2
pointer: 400924  :b8 0 0 0 0 e8 b2 fd
pointer: 400925  :0 0 0 0 e8 b2 fd ff
pointer: 400926  :0 0 0 e8 b2 fd ff ff
pointer: 400927  :0 0 e8 b2 fd ff ff e8
pointer: 400928  :0 e8 b2 fd ff ff e8 cd
pointer: 400929  :e8 b2 fd ff ff e8 cd fd
pointer: 40092a  :b2 fd ff ff e8 cd fd ff
pointer: 40092b  :fd ff ff e8 cd fd ff ff
pointer: 40092c  :ff ff e8 cd fd ff ff 88
pointer: 40092d  :ff e8 cd fd ff ff 88 45
pointer: 40092e  :e8 cd fd ff ff 88 45 f7
pointer: 40092f  :cd fd ff ff 88 45 f7 e8
```

```
pointer: 400930  :fd ff ff 88 45 f7 e8 c5
pointer: 400931  :ff ff 88 45 f7 e8 c5 fd
pointer: 400932  :ff 88 45 f7 e8 c5 fd ff
pointer: 400933  :88 45 f7 e8 c5 fd ff ff
pointer: 400934  :45 f7 e8 c5 fd ff ff 80
pointer: 400935  :f7 e8 c5 fd ff ff 80 7d
pointer: 400936  :e8 c5 fd ff ff 80 7d f7
pointer: 400937  :c5 fd ff ff 80 7d f7 6e
pointer: 400938  :fd ff ff 80 7d f7 6e f
pointer: 400939  :ff ff 80 7d f7 6e f 84
pointer: 40093a  :ff 80 7d f7 6e f 84 3e
pointer: 40093b  :80 7d f7 6e f 84 3e ff
pointer: 40093c  :7d f7 6e f 84 3e ff ff
pointer: 40093d  :f7 6e f 84 3e ff ff ff
pointer: 40093e  :6e f 84 3e ff ff ff b8
pointer: 40093f  :f 84 3e ff ff ff b8 0
pointer: 400940  :84 3e ff ff ff b8 0 0
pointer: 400941  :3e ff ff ff b8 0 0 0
pointer: 400942  :ff ff ff b8 0 0 0 0
pointer: 400943  :ff ff b8 0 0 0 0 c9
pointer: 400944  :ff b8 0 0 0 0 c9 c3
pointer: 400945  :b8 0 0 0 0 c9 c3 55
pointer: 400946  :0 0 0 0 c9 c3 55 48
pointer: 400947  :0 0 0 c9 c3 55 48 89
pointer: 400948  :0 0 c9 c3 55 48 89 e5
pointer: 400949  :0 c9 c3 55 48 89 e5 48
pointer: 40094a  :c9 c3 55 48 89 e5 48 8d
pointer: 40094b  :c3 55 48 89 e5 48 8d 3d
pointer: 40094c  :55 48 89 e5 48 8d 3d 35
pointer: 40094d  :48 89 e5 48 8d 3d 35 6
pointer: 40094e  :89 e5 48 8d 3d 35 6 0
pointer: 40094f  :e5 48 8d 3d 35 6 0 0
pointer: 400950  :48 8d 3d 35 6 0 0 e8
pointer: 400951  :8d 3d 35 6 0 0 e8 44
```

```
pointer: 400952  :3d 35 6 0 0 e8 44 fd
pointer: 400953  :35 6 0 0 e8 44 fd ff
pointer: 400954  :6 0 0 e8 44 fd ff ff
pointer: 400955  :0 0 e8 44 fd ff ff 48
pointer: 400956  :0 e8 44 fd ff ff 48 8d
pointer: 400957  :e8 44 fd ff ff 48 8d 5
pointer: 400958  :44 fd ff ff 48 8d 5 5d
pointer: 400959  :fd ff ff 48 8d 5 5d 37
pointer: 40095a  :ff ff 48 8d 5 5d 37 20
pointer: 40095b  :ff 48 8d 5 5d 37 20 0
pointer: 40095c  :48 8d 5 5d 37 20 0 48
pointer: 40095d  :8d 5 5d 37 20 0 48 89
pointer: 40095e  :5 5d 37 20 0 48 89 c6
pointer: 40095f  :5d 37 20 0 48 89 c6 48
pointer: 400960  :37 20 0 48 89 c6 48 8d
pointer: 400961  :20 0 48 89 c6 48 8d 3d
pointer: 400962  :0 48 89 c6 48 8d 3d 25
pointer: 400963  :48 89 c6 48 8d 3d 25 6
pointer: 400964  :89 c6 48 8d 3d 25 6 0
pointer: 400965  :c6 48 8d 3d 25 6 0 0
pointer: 400966  :48 8d 3d 25 6 0 0 b8
pointer: 400967  :8d 3d 25 6 0 0 b8 0
pointer: 400968  :3d 25 6 0 0 b8 0 0
pointer: 400969  :25 6 0 0 b8 0 0 0
pointer: 40096a  :6 0 0 b8 0 0 0 0
pointer: 40096b  :0 0 b8 0 0 0 0 e8
pointer: 40096c  :0 b8 0 0 0 0 e8 69
pointer: 40096d  :b8 0 0 0 0 e8 69 fd
pointer: 40096e  :0 0 0 0 e8 69 fd ff
pointer: 40096f  :0 0 0 e8 69 fd ff ff
pointer: 400970  :0 0 e8 69 fd ff ff 48
pointer: 400971  :0 e8 69 fd ff ff 48 8d
pointer: 400972  :e8 69 fd ff ff 48 8d 5
pointer: 400973  :69 fd ff ff 48 8d 5 a2
```

```
pointer: 400974  :fd ff ff 48 8d 5 a2 27
pointer: 400975  :ff ff 48 8d 5 a2 27 20
pointer: 400976  :ff 48 8d 5 a2 27 20 0
pointer: 400977  :48 8d 5 a2 27 20 0 48
pointer: 400978  :8d 5 a2 27 20 0 48 89
pointer: 400979  :5 a2 27 20 0 48 89 c6
pointer: 40097a  :a2 27 20 0 48 89 c6 48
pointer: 40097b  :27 20 0 48 89 c6 48 8d
pointer: 40097c  :20 0 48 89 c6 48 8d 3d
pointer: 40097d  :0 48 89 c6 48 8d 3d 1c
pointer: 40097e  :48 89 c6 48 8d 3d 1c 6
pointer: 40097f  :89 c6 48 8d 3d 1c 6 0
pointer: 400980  :c6 48 8d 3d 1c 6 0 0
pointer: 400981  :48 8d 3d 1c 6 0 0 b8
pointer: 400982  :8d 3d 1c 6 0 0 b8 0
pointer: 400983  :3d 1c 6 0 0 b8 0 0
pointer: 400984  :1c 6 0 0 b8 0 0 0
pointer: 400985  :6 0 0 b8 0 0 0 0
pointer: 400986  :0 0 b8 0 0 0 0 e8
pointer: 400987  :0 b8 0 0 0 0 e8 4e
pointer: 400988  :b8 0 0 0 0 e8 4e fd
pointer: 400989  :0 0 0 0 e8 4e fd ff
pointer: 40098a  :0 0 0 e8 4e fd ff ff
pointer: 40098b  :0 0 e8 4e fd ff ff 48
pointer: 40098c  :0 e8 4e fd ff ff 48 8d
pointer: 40098d  :e8 4e fd ff ff 48 8d 5
pointer: 40098e  :4e fd ff ff 48 8d 5 c7
pointer: 40098f  :fd ff ff 48 8d 5 c7 46
pointer: 400990  :ff ff 48 8d 5 c7 46 20
pointer: 400991  :ff 48 8d 5 c7 46 20 0
pointer: 400992  :48 8d 5 c7 46 20 0 48
pointer: 400993  :8d 5 c7 46 20 0 48 89
pointer: 400994  :5 c7 46 20 0 48 89 c6
pointer: 400995  :c7 46 20 0 48 89 c6 48
```

```
pointer: 400996  :46 20 0 48 89 c6 48 8d
pointer: 400997  :20 0 48 89 c6 48 8d 3d
pointer: 400998  :0 48 89 c6 48 8d 3d 13
pointer: 400999  :48 89 c6 48 8d 3d 13 6
pointer: 40099a  :89 c6 48 8d 3d 13 6 0
pointer: 40099b  :c6 48 8d 3d 13 6 0 0
pointer: 40099c  :48 8d 3d 13 6 0 0 b8
pointer: 40099d  :8d 3d 13 6 0 0 b8 0
pointer: 40099e  :3d 13 6 0 0 b8 0 0
pointer: 40099f  :13 6 0 0 b8 0 0 0
pointer: 4009a0  :6 0 0 b8 0 0 0 0
pointer: 4009a1  :0 0 b8 0 0 0 0 e8
pointer: 4009a2  :0 b8 0 0 0 0 e8 33
pointer: 4009a3  :b8 0 0 0 0 e8 33 fd
pointer: 4009a4  :0 0 0 0 e8 33 fd ff
pointer: 4009a5  :0 0 0 e8 33 fd ff ff
pointer: 4009a6  :0 0 e8 33 fd ff ff 48
pointer: 4009a7  :0 e8 33 fd ff ff 48 8d
pointer: 4009a8  :e8 33 fd ff ff 48 8d 5
pointer: 4009a9  :33 fd ff ff 48 8d 5 cc
pointer: 4009aa  :fd ff ff 48 8d 5 cc 17
pointer: 4009ab  :ff ff 48 8d 5 cc 17 20
pointer: 4009ac  :ff 48 8d 5 cc 17 20 0
pointer: 4009ad  :48 8d 5 cc 17 20 0 48
pointer: 4009ae  :8d 5 cc 17 20 0 48 89
pointer: 4009af  :5 cc 17 20 0 48 89 c6
pointer: 4009b0  :cc 17 20 0 48 89 c6 48
pointer: 4009b1  :17 20 0 48 89 c6 48 8d
pointer: 4009b2  :20 0 48 89 c6 48 8d 3d
pointer: 4009b3  :0 48 89 c6 48 8d 3d a
pointer: 4009b4  :48 89 c6 48 8d 3d a 6
pointer: 4009b5  :89 c6 48 8d 3d a 6 0
pointer: 4009b6  :c6 48 8d 3d a 6 0 0
pointer: 4009b7  :48 8d 3d a 6 0 0 b8
```

```
pointer: 4009b8  :8d 3d a 6 0 0 b8 0
pointer: 4009b9  :3d a 6 0 0 b8 0 0
pointer: 4009ba  :a 6 0 0 b8 0 0 0
pointer: 4009bb  :6 0 0 b8 0 0 0 0
pointer: 4009bc  :0 0 b8 0 0 0 0 e8
pointer: 4009bd  :0 b8 0 0 0 0 e8 18
pointer: 4009be  :b8 0 0 0 0 e8 18 fd
pointer: 4009bf  :0 0 0 0 e8 18 fd ff
pointer: 4009c0  :0 0 0 e8 18 fd ff ff
pointer: 4009c1  :0 0 e8 18 fd ff ff 48
pointer: 4009c2  :0 e8 18 fd ff ff 48 8d
pointer: 4009c3  :e8 18 fd ff ff 48 8d 5
pointer: 4009c4  :18 fd ff ff 48 8d 5 f1
pointer: 4009c5  :fd ff ff 48 8d 5 f1 16
pointer: 4009c6  :ff ff 48 8d 5 f1 16 20
pointer: 4009c7  :ff 48 8d 5 f1 16 20 0
pointer: 4009c8  :48 8d 5 f1 16 20 0 48
pointer: 4009c9  :8d 5 f1 16 20 0 48 89
pointer: 4009ca  :5 f1 16 20 0 48 89 c6
pointer: 4009cb  :f1 16 20 0 48 89 c6 48
pointer: 4009cc  :16 20 0 48 89 c6 48 8d
pointer: 4009cd  :20 0 48 89 c6 48 8d 3d
pointer: 4009ce  :0 48 89 c6 48 8d 3d 1
pointer: 4009cf  :48 89 c6 48 8d 3d 1 6
pointer: 4009d0  :89 c6 48 8d 3d 1 6 0
pointer: 4009d1  :c6 48 8d 3d 1 6 0 0
pointer: 4009d2  :48 8d 3d 1 6 0 0 b8
pointer: 4009d3  :8d 3d 1 6 0 0 b8 0
pointer: 4009d4  :3d 1 6 0 0 b8 0 0
pointer: 4009d5  :1 6 0 0 b8 0 0 0
pointer: 4009d6  :6 0 0 b8 0 0 0 0
pointer: 4009d7  :0 0 b8 0 0 0 0 e8
pointer: 4009d8  :0 b8 0 0 0 0 e8 fd
pointer: 4009d9  :b8 0 0 0 0 e8 fd fc
```

```
pointer: 4009da  :0 0 0 0 e8 fd fc ff
pointer: 4009db  :0 0 0 e8 fd fc ff ff
pointer: 4009dc  :0 0 e8 fd fc ff ff 48
pointer: 4009dd  :0 e8 fd fc ff ff 48 8d
pointer: 4009de  :e8 fd fc ff ff 48 8d 5
pointer: 4009df  :fd fc ff ff 48 8d 5 f6
pointer: 4009e0  :fc ff ff 48 8d 5 f6 16
pointer: 4009e1  :ff ff 48 8d 5 f6 16 20
pointer: 4009e2  :ff 48 8d 5 f6 16 20 0
pointer: 4009e3  :48 8d 5 f6 16 20 0 48
pointer: 4009e4  :8d 5 f6 16 20 0 48 89
pointer: 4009e5  :5 f6 16 20 0 48 89 c6
pointer: 4009e6  :f6 16 20 0 48 89 c6 48
pointer: 4009e7  :16 20 0 48 89 c6 48 8d
pointer: 4009e8  :20 0 48 89 c6 48 8d 3d
pointer: 4009e9  :0 48 89 c6 48 8d 3d f6
pointer: 4009ea  :48 89 c6 48 8d 3d f6 5
pointer: 4009eb  :89 c6 48 8d 3d f6 5 0
pointer: 4009ec  :c6 48 8d 3d f6 5 0 0
pointer: 4009ed  :48 8d 3d f6 5 0 0 b8
pointer: 4009ee  :8d 3d f6 5 0 0 b8 0
pointer: 4009ef  :3d f6 5 0 0 b8 0 0
pointer: 4009f0  :f6 5 0 0 b8 0 0 0
pointer: 4009f1  :5 0 0 b8 0 0 0 0
pointer: 4009f2  :0 0 b8 0 0 0 0 e8
pointer: 4009f3  :0 b8 0 0 0 0 e8 e2
pointer: 4009f4  :b8 0 0 0 0 e8 e2 fc
pointer: 4009f5  :0 0 0 0 e8 e2 fc ff
pointer: 4009f6  :0 0 0 e8 e2 fc ff ff
pointer: 4009f7  :0 0 e8 e2 fc ff ff 48
pointer: 4009f8  :0 e8 e2 fc ff ff 48 8d
pointer: 4009f9  :e8 e2 fc ff ff 48 8d 5
pointer: 4009fa  :e2 fc ff ff 48 8d 5 fb
pointer: 4009fb  :fc ff ff 48 8d 5 fb 16
```

```
pointer: 4009fc  :ff ff 48 8d 5 fb 16 20
pointer: 4009fd  :ff 48 8d 5 fb 16 20 0
pointer: 4009fe  :48 8d 5 fb 16 20 0 48
pointer: 4009ff  :8d 5 fb 16 20 0 48 89
pointer: 400a00  :5 fb 16 20 0 48 89 c6
pointer: 400a01  :fb 16 20 0 48 89 c6 48
pointer: 400a02  :16 20 0 48 89 c6 48 8d
pointer: 400a03  :20 0 48 89 c6 48 8d 3d
pointer: 400a04  :0 48 89 c6 48 8d 3d eb
pointer: 400a05  :48 89 c6 48 8d 3d eb 5
pointer: 400a06  :89 c6 48 8d 3d eb 5 0
pointer: 400a07  :c6 48 8d 3d eb 5 0 0
pointer: 400a08  :48 8d 3d eb 5 0 0 b8
pointer: 400a09  :8d 3d eb 5 0 0 b8 0
pointer: 400a0a  :3d eb 5 0 0 b8 0 0
pointer: 400a0b  :eb 5 0 0 b8 0 0 0
pointer: 400a0c  :5 0 0 b8 0 0 0 0
pointer: 400a0d  :0 0 b8 0 0 0 0 e8
pointer: 400a0e  :0 b8 0 0 0 0 e8 c7
pointer: 400a0f  :b8 0 0 0 0 e8 c7 fc
pointer: 400a10  :0 0 0 0 e8 c7 fc ff
pointer: 400a11  :0 0 0 e8 c7 fc ff ff
pointer: 400a12  :0 0 e8 c7 fc ff ff 48
pointer: 400a13  :0 e8 c7 fc ff ff 48 8d
pointer: 400a14  :e8 c7 fc ff ff 48 8d 5
pointer: 400a15  :c7 fc ff ff 48 8d 5 0
pointer: 400a16  :fc ff ff 48 8d 5 0 17
pointer: 400a17  :ff ff 48 8d 5 0 17 20
pointer: 400a18  :ff 48 8d 5 0 17 20 0
pointer: 400a19  :48 8d 5 0 17 20 0 48
pointer: 400a1a  :8d 5 0 17 20 0 48 89
pointer: 400a1b  :5 0 17 20 0 48 89 c6
pointer: 400a1c  :0 17 20 0 48 89 c6 48
pointer: 400a1d  :17 20 0 48 89 c6 48 8d
```

```
pointer: 400a1e  :20 0 48 89 c6 48 8d 3d
pointer: 400a1f  :0 48 89 c6 48 8d 3d e0
pointer: 400a20  :48 89 c6 48 8d 3d e0 5
pointer: 400a21  :89 c6 48 8d 3d e0 5 0
pointer: 400a22  :c6 48 8d 3d e0 5 0 0
pointer: 400a23  :48 8d 3d e0 5 0 0 b8
pointer: 400a24  :8d 3d e0 5 0 0 b8 0
pointer: 400a25  :3d e0 5 0 0 b8 0 0
pointer: 400a26  :e0 5 0 0 b8 0 0 0
pointer: 400a27  :5 0 0 b8 0 0 0 0
pointer: 400a28  :0 0 b8 0 0 0 0 e8
pointer: 400a29  :0 b8 0 0 0 0 e8 ac
pointer: 400a2a  :b8 0 0 0 0 e8 ac fc
pointer: 400a2b  :0 0 0 0 e8 ac fc ff
pointer: 400a2c  :0 0 0 e8 ac fc ff ff
pointer: 400a2d  :0 0 e8 ac fc ff ff 48
pointer: 400a2e  :0 e8 ac fc ff ff 48 8d
pointer: 400a2f  :e8 ac fc ff ff 48 8d 3d
pointer: 400a30  :ac fc ff ff 48 8d 3d df
pointer: 400a31  :fc ff ff 48 8d 3d df 5
pointer: 400a32  :ff ff 48 8d 3d df 5 0
pointer: 400a33  :ff 48 8d 3d df 5 0 0
pointer: 400a34  :48 8d 3d df 5 0 0 e8
pointer: 400a35  :8d 3d df 5 0 0 e8 60
pointer: 400a36  :3d df 5 0 0 e8 60 fc
pointer: 400a37  :df 5 0 0 e8 60 fc ff
pointer: 400a38  :5 0 0 e8 60 fc ff ff
pointer: 400a39  :0 0 e8 60 fc ff ff 48
pointer: 400a3a  :0 e8 60 fc ff ff 48 8d
pointer: 400a3b  :e8 60 fc ff ff 48 8d 5
pointer: 400a3c  :60 fc ff ff 48 8d 5 f0
pointer: 400a3d  :fc ff ff 48 8d 5 f0 fd
pointer: 400a3e  :ff ff 48 8d 5 f0 fd ff
pointer: 400a3f  :ff 48 8d 5 f0 fd ff ff
```

```
pointer: 400a40  :48 8d 5 f0 fd ff ff 48
pointer: 400a41  :8d 5 f0 fd ff ff 48 89
pointer: 400a42  :5 f0 fd ff ff 48 89 c6
pointer: 400a43  :f0 fd ff ff 48 89 c6 48
pointer: 400a44  :fd ff ff 48 89 c6 48 8d
pointer: 400a45  :ff ff 48 89 c6 48 8d 3d
pointer: 400a46  :ff 48 89 c6 48 8d 3d d4
pointer: 400a47  :48 89 c6 48 8d 3d d4 5
pointer: 400a48  :89 c6 48 8d 3d d4 5 0
pointer: 400a49  :c6 48 8d 3d d4 5 0 0
pointer: 400a4a  :48 8d 3d d4 5 0 0 b8
pointer: 400a4b  :8d 3d d4 5 0 0 b8 0
pointer: 400a4c  :3d d4 5 0 0 b8 0 0
pointer: 400a4d  :d4 5 0 0 b8 0 0 0
pointer: 400a4e  :5 0 0 b8 0 0 0 0
pointer: 400a4f  :0 0 b8 0 0 0 0 e8
pointer: 400a50  :0 b8 0 0 0 0 e8 85
pointer: 400a51  :b8 0 0 0 0 e8 85 fc
pointer: 400a52  :0 0 0 0 e8 85 fc ff
pointer: 400a53  :0 0 0 e8 85 fc ff ff
pointer: 400a54  :0 0 e8 85 fc ff ff 48
pointer: 400a55  :0 e8 85 fc ff ff 48 8d
pointer: 400a56  :e8 85 fc ff ff 48 8d 5
pointer: 400a57  :85 fc ff ff 48 8d 5 ea
pointer: 400a58  :fc ff ff 48 8d 5 ea fe
pointer: 400a59  :ff ff 48 8d 5 ea fe ff
pointer: 400a5a  :ff 48 8d 5 ea fe ff ff
pointer: 400a5b  :48 8d 5 ea fe ff ff 48
pointer: 400a5c  :8d 5 ea fe ff ff 48 89
pointer: 400a5d  :5 ea fe ff ff 48 89 c6
pointer: 400a5e  :ea fe ff ff 48 89 c6 48
pointer: 400a5f  :fe ff ff 48 89 c6 48 8d
pointer: 400a60  :ff ff 48 89 c6 48 8d 3d
pointer: 400a61  :ff 48 89 c6 48 8d 3d ca
```

```
pointer: 400a62  :48 89 c6 48 8d 3d ca 5

pointer: 400a63  :89 c6 48 8d 3d ca 5 0

pointer: 400a64  :c6 48 8d 3d ca 5 0 0

pointer: 400a65  :48 8d 3d ca 5 0 0 b8

pointer: 400a66  :8d 3d ca 5 0 0 b8 0

pointer: 400a67  :3d ca 5 0 0 b8 0 0

pointer: 400a68  :ca 5 0 0 b8 0 0 0

pointer: 400a69  :5 0 0 b8 0 0 0 0

pointer: 400a6a  :0 0 b8 0 0 0 0 e8

pointer: 400a6b  :0 b8 0 0 0 0 e8 6a

pointer: 400a6c  :b8 0 0 0 0 e8 6a fc

pointer: 400a6d  :0 0 0 0 e8 6a fc ff

pointer: 400a6e  :0 0 0 e8 6a fc ff ff

pointer: 400a6f  :0 0 e8 6a fc ff ff 48

pointer: 400a70  :0 e8 6a fc ff ff 48 8d

pointer: 400a71  :e8 6a fc ff ff 48 8d 5

pointer: 400a72  :6a fc ff ff 48 8d 5 83

pointer: 400a73  :fc ff ff 48 8d 5 83 0

pointer: 400a74  :ff ff 48 8d 5 83 0 0

pointer: 400a75  :ff 48 8d 5 83 0 0 0

pointer: 400a76  :48 8d 5 83 0 0 0 48

pointer: 400a77  :8d 5 83 0 0 0 48 89

pointer: 400a78  :5 83 0 0 0 48 89 c6

pointer: 400a79  :83 0 0 0 48 89 c6 48

pointer: 400a7a  :0 0 0 48 89 c6 48 8d

pointer: 400a7b  :0 0 48 89 c6 48 8d 3d

pointer: 400a7c  :0 48 89 c6 48 8d 3d c1

pointer: 400a7d  :48 89 c6 48 8d 3d c1 5

pointer: 400a7e  :89 c6 48 8d 3d c1 5 0

pointer: 400a7f  :c6 48 8d 3d c1 5 0 0

pointer: 400a80  :48 8d 3d c1 5 0 0 b8

pointer: 400a81  :8d 3d c1 5 0 0 b8 0

pointer: 400a82  :3d c1 5 0 0 b8 0 0

pointer: 400a83  :c1 5 0 0 b8 0 0 0
```

```
pointer: 400a84  :5 0 0 b8 0 0 0 0
pointer: 400a85  :0 0 b8 0 0 0 0 e8
pointer: 400a86  :0 b8 0 0 0 0 e8 4f
pointer: 400a87  :b8 0 0 0 0 e8 4f fc
pointer: 400a88  :0 0 0 0 e8 4f fc ff
pointer: 400a89  :0 0 0 e8 4f fc ff ff
pointer: 400a8a  :0 0 e8 4f fc ff ff 48
pointer: 400a8b  :0 e8 4f fc ff ff 48 8d
pointer: 400a8c  :e8 4f fc ff ff 48 8d 5
pointer: 400a8d  :4f fc ff ff 48 8d 5 f1
pointer: 400a8e  :fc ff ff 48 8d 5 f1 0
pointer: 400a8f  :ff ff 48 8d 5 f1 0 0
pointer: 400a90  :ff 48 8d 5 f1 0 0 0
pointer: 400a91  :48 8d 5 f1 0 0 0 48
pointer: 400a92  :8d 5 f1 0 0 0 48 89
pointer: 400a93  :5 f1 0 0 0 48 89 c6
pointer: 400a94  :f1 0 0 0 48 89 c6 48
pointer: 400a95  :0 0 0 48 89 c6 48 8d
pointer: 400a96  :0 0 48 89 c6 48 8d 3d
pointer: 400a97  :0 48 89 c6 48 8d 3d b8
pointer: 400a98  :48 89 c6 48 8d 3d b8 5
pointer: 400a99  :89 c6 48 8d 3d b8 5 0
pointer: 400a9a  :c6 48 8d 3d b8 5 0 0
pointer: 400a9b  :48 8d 3d b8 5 0 0 b8
pointer: 400a9c  :8d 3d b8 5 0 0 b8 0
pointer: 400a9d  :3d b8 5 0 0 b8 0 0
pointer: 400a9e  :b8 5 0 0 b8 0 0 0
pointer: 400a9f  :5 0 0 b8 0 0 0 0
pointer: 400aa0  :0 0 b8 0 0 0 0 e8
pointer: 400aa1  :0 b8 0 0 0 0 e8 34
pointer: 400aa2  :b8 0 0 0 0 e8 34 fc
pointer: 400aa3  :0 0 0 0 e8 34 fc ff
pointer: 400aa4  :0 0 0 e8 34 fc ff ff
pointer: 400aa5  :0 0 e8 34 fc ff ff 48
```

```
pointer: 400aa6  :0 e8 34 fc ff ff 48 8d
pointer: 400aa7  :e8 34 fc ff ff 48 8d 5
pointer: 400aa8  :34 fc ff ff 48 8d 5 f8
pointer: 400aa9  :fc ff ff 48 8d 5 f8 0
pointer: 400aaa  :ff ff 48 8d 5 f8 0 0
pointer: 400aab  :ff 48 8d 5 f8 0 0 0
pointer: 400aac  :48 8d 5 f8 0 0 0 48
pointer: 400aad  :8d 5 f8 0 0 0 48 89
pointer: 400aae  :5 f8 0 0 0 48 89 c6
pointer: 400aaf  :f8 0 0 0 48 89 c6 48
pointer: 400ab0  :0 0 0 48 89 c6 48 8d
pointer: 400ab1  :0 0 48 89 c6 48 8d 3d
pointer: 400ab2  :0 48 89 c6 48 8d 3d af
pointer: 400ab3  :48 89 c6 48 8d 3d af 5
pointer: 400ab4  :89 c6 48 8d 3d af 5 0
pointer: 400ab5  :c6 48 8d 3d af 5 0 0
pointer: 400ab6  :48 8d 3d af 5 0 0 b8
pointer: 400ab7  :8d 3d af 5 0 0 b8 0
pointer: 400ab8  :3d af 5 0 0 b8 0 0
pointer: 400ab9  :af 5 0 0 b8 0 0 0
pointer: 400aba  :5 0 0 b8 0 0 0 0
pointer: 400abb  :0 0 b8 0 0 0 0 e8
pointer: 400abc  :0 b8 0 0 0 0 e8 19
pointer: 400abd  :b8 0 0 0 0 e8 19 fc
pointer: 400abe  :0 0 0 0 e8 19 fc ff
pointer: 400abf  :0 0 0 e8 19 fc ff ff
pointer: 400ac0  :0 0 e8 19 fc ff ff 48
pointer: 400ac1  :0 e8 19 fc ff ff 48 8d
pointer: 400ac2  :e8 19 fc ff ff 48 8d 5
pointer: 400ac3  :19 fc ff ff 48 8d 5 a0
pointer: 400ac4  :fc ff ff 48 8d 5 a0 1
pointer: 400ac5  :ff ff 48 8d 5 a0 1 0
pointer: 400ac6  :ff 48 8d 5 a0 1 0 0
pointer: 400ac7  :48 8d 5 a0 1 0 0 48
```

```
pointer: 400ac8  :8d 5 a0 1 0 0 48 89
pointer: 400ac9  :5 a0 1 0 0 48 89 c6
pointer: 400aca  :a0 1 0 0 48 89 c6 48
pointer: 400acb  :1 0 0 48 89 c6 48 8d
pointer: 400acc  :0 0 48 89 c6 48 8d 3d
pointer: 400acd  :0 48 89 c6 48 8d 3d a6
pointer: 400ace  :48 89 c6 48 8d 3d a6 5
pointer: 400acf  :89 c6 48 8d 3d a6 5 0
pointer: 400ad0  :c6 48 8d 3d a6 5 0 0
pointer: 400ad1  :48 8d 3d a6 5 0 0 b8
pointer: 400ad2  :8d 3d a6 5 0 0 b8 0
pointer: 400ad3  :3d a6 5 0 0 b8 0 0
pointer: 400ad4  :a6 5 0 0 b8 0 0 0
pointer: 400ad5  :5 0 0 b8 0 0 0 0
pointer: 400ad6  :0 0 b8 0 0 0 0 e8
pointer: 400ad7  :0 b8 0 0 0 0 e8 fe
pointer: 400ad8  :b8 0 0 0 0 e8 fe fb
pointer: 400ad9  :0 0 0 0 e8 fe fb ff
pointer: 400ada  :0 0 0 e8 fe fb ff ff
pointer: 400adb  :0 0 e8 fe fb ff ff 48
pointer: 400adc  :0 e8 fe fb ff ff 48 8d
pointer: 400add  :e8 fe fb ff ff 48 8d 5
pointer: 400ade  :fe fb ff ff 48 8d 5 78
pointer: 400adf  :fb ff ff 48 8d 5 78 2
pointer: 400ae0  :ff ff 48 8d 5 78 2 0
pointer: 400ae1  :ff 48 8d 5 78 2 0 0
pointer: 400ae2  :48 8d 5 78 2 0 0 48
pointer: 400ae3  :8d 5 78 2 0 0 48 89
pointer: 400ae4  :5 78 2 0 0 48 89 c6
pointer: 400ae5  :78 2 0 0 48 89 c6 48
pointer: 400ae6  :2 0 0 48 89 c6 48 8d
pointer: 400ae7  :0 0 48 89 c6 48 8d 3d
pointer: 400ae8  :0 48 89 c6 48 8d 3d 9d
pointer: 400ae9  :48 89 c6 48 8d 3d 9d 5
```

```
pointer: 400aea  :89 c6 48 8d 3d 9d 5 0
pointer: 400aeb  :c6 48 8d 3d 9d 5 0 0
pointer: 400aec  :48 8d 3d 9d 5 0 0 b8
pointer: 400aed  :8d 3d 9d 5 0 0 b8 0
pointer: 400aee  :3d 9d 5 0 0 b8 0 0
pointer: 400aef  :9d 5 0 0 b8 0 0 0
pointer: 400af0  :5 0 0 b8 0 0 0 0
pointer: 400af1  :0 0 b8 0 0 0 0 e8
pointer: 400af2  :0 b8 0 0 0 0 e8 e3
pointer: 400af3  :b8 0 0 0 0 e8 e3 fb
pointer: 400af4  :0 0 0 0 e8 e3 fb ff
pointer: 400af5  :0 0 0 e8 e3 fb ff ff
pointer: 400af6  :0 0 e8 e3 fb ff ff 90
pointer: 400af7  :0 e8 e3 fb ff ff 90 5d
pointer: 400af8  :e8 e3 fb ff ff 90 5d c3
pointer: 400af9  :e3 fb ff ff 90 5d c3 55
pointer: 400afa  :fb ff ff 90 5d c3 55 48
pointer: 400afb  :ff ff 90 5d c3 55 48 89
pointer: 400afc  :ff 90 5d c3 55 48 89 e5
pointer: 400afd  :90 5d c3 55 48 89 e5 48
pointer: 400afe  :5d c3 55 48 89 e5 48 83
pointer: 400aff  :c3 55 48 89 e5 48 83 ec
pointer: 400b00  :55 48 89 e5 48 83 ec 20
pointer: 400b01  :48 89 e5 48 83 ec 20 bf
pointer: 400b02  :89 e5 48 83 ec 20 bf 80
pointer: 400b03  :e5 48 83 ec 20 bf 80 84
pointer: 400b04  :48 83 ec 20 bf 80 84 1e
pointer: 400b05  :83 ec 20 bf 80 84 1e 0
pointer: 400b06  :ec 20 bf 80 84 1e 0 e8
pointer: 400b07  :20 bf 80 84 1e 0 e8 fe
pointer: 400b08  :bf 80 84 1e 0 e8 fe fb
pointer: 400b09  :80 84 1e 0 e8 fe fb ff
pointer: 400b0a  :84 1e 0 e8 fe fb ff ff
pointer: 400b0b  :1e 0 e8 fe fb ff ff 48
```

```
pointer: 400b0c  :0 e8 fe fb ff ff 48 89
pointer: 400b0d  :e8 fe fb ff ff 48 89 45
pointer: 400b0e  :fe fb ff ff 48 89 45 e8
pointer: 400b0f  :fb ff ff 48 89 45 e8 bf
pointer: 400b10  :ff ff 48 89 45 e8 bf 40
pointer: 400b11  :ff 48 89 45 e8 bf 40 42
pointer: 400b12  :48 89 45 e8 bf 40 42 f
pointer: 400b13  :89 45 e8 bf 40 42 f 0
pointer: 400b14  :45 e8 bf 40 42 f 0 e8
pointer: 400b15  :e8 bf 40 42 f 0 e8 f0
pointer: 400b16  :bf 40 42 f 0 e8 f0 fb
pointer: 400b17  :40 42 f 0 e8 f0 fb ff
pointer: 400b18  :42 f 0 e8 f0 fb ff ff
pointer: 400b19  :f 0 e8 f0 fb ff ff 48
pointer: 400b1a  :0 e8 f0 fb ff ff 48 89
pointer: 400b1b  :e8 f0 fb ff ff 48 89 45
pointer: 400b1c  :f0 fb ff ff 48 89 45 f0
pointer: 400b1d  :fb ff ff 48 89 45 f0 bf
pointer: 400b1e  :ff ff 48 89 45 f0 bf 40
pointer: 400b1f  :ff 48 89 45 f0 bf 40 42
pointer: 400b20  :48 89 45 f0 bf 40 42 f
pointer: 400b21  :89 45 f0 bf 40 42 f 0
pointer: 400b22  :45 f0 bf 40 42 f 0 e8
pointer: 400b23  :f0 bf 40 42 f 0 e8 e2
pointer: 400b24  :bf 40 42 f 0 e8 e2 fb
pointer: 400b25  :40 42 f 0 e8 e2 fb ff
pointer: 400b26  :42 f 0 e8 e2 fb ff ff
pointer: 400b27  :f 0 e8 e2 fb ff ff 48
pointer: 400b28  :0 e8 e2 fb ff ff 48 89
pointer: 400b29  :e8 e2 fb ff ff 48 89 45
pointer: 400b2a  :e2 fb ff ff 48 89 45 f8
pointer: 400b2b  :fb ff ff 48 89 45 f8 48
pointer: 400b2c  :ff ff 48 89 45 f8 48 8d
pointer: 400b2d  :ff 48 89 45 f8 48 8d 3d
```

```
pointer: 400b2e  :48 89 45 f8 48 8d 3d 69
pointer: 400b2f :89 45 f8 48 8d 3d 69 5
pointer: 400b30  :45 f8 48 8d 3d 69 5 0
pointer: 400b31  :f8 48 8d 3d 69 5 0 0
pointer: 400b32  :48 8d 3d 69 5 0 0 e8
pointer: 400b33  :8d 3d 69 5 0 0 e8 62
pointer: 400b34  :3d 69 5 0 0 e8 62 fb
pointer: 400b35  :69 5 0 0 e8 62 fb ff
pointer: 400b36  :5 0 0 e8 62 fb ff ff
pointer: 400b37  :0 0 e8 62 fb ff ff 48
pointer: 400b38  :0 e8 62 fb ff ff 48 8b
pointer: 400b39  :e8 62 fb ff ff 48 8b 45
pointer: 400b3a  :62 fb ff ff 48 8b 45 e8
pointer: 400b3b  :fb ff ff 48 8b 45 e8 48
pointer: 400b3c  :ff ff 48 8b 45 e8 48 89
pointer: 400b3d  :ff 48 8b 45 e8 48 89 c6
pointer: 400b3e  :48 8b 45 e8 48 89 c6 48
pointer: 400b3f :8b 45 e8 48 89 c6 48 8d
pointer: 400b40  :45 e8 48 89 c6 48 8d 3d
pointer: 400b41  :e8 48 89 c6 48 8d 3d 68
pointer: 400b42  :48 89 c6 48 8d 3d 68 5
pointer: 400b43  :89 c6 48 8d 3d 68 5 0
pointer: 400b44  :c6 48 8d 3d 68 5 0 0
pointer: 400b45  :48 8d 3d 68 5 0 0 b8
pointer: 400b46  :8d 3d 68 5 0 0 b8 0
pointer: 400b47  :3d 68 5 0 0 b8 0 0
pointer: 400b48  :68 5 0 0 b8 0 0 0
pointer: 400b49  :5 0 0 b8 0 0 0 0
pointer: 400b4a  :0 0 b8 0 0 0 0 e8
pointer: 400b4b  :0 b8 0 0 0 0 e8 8a
pointer: 400b4c  :b8 0 0 0 0 e8 8a fb
pointer: 400b4d  :0 0 0 0 e8 8a fb ff
pointer: 400b4e  :0 0 0 e8 8a fb ff ff
pointer: 400b4f :0 0 e8 8a fb ff ff 48
```

```
pointer: 400b50  :0 e8 8a fb ff ff 48 8b
pointer: 400b51  :e8 8a fb ff ff 48 8b 45
pointer: 400b52  :8a fb ff ff 48 8b 45 f0
pointer: 400b53  :fb ff ff 48 8b 45 f0 48
pointer: 400b54  :ff ff 48 8b 45 f0 48 89
pointer: 400b55  :ff 48 8b 45 f0 48 89 c6
pointer: 400b56  :48 8b 45 f0 48 89 c6 48
pointer: 400b57  :8b 45 f0 48 89 c6 48 8d
pointer: 400b58  :45 f0 48 89 c6 48 8d 3d
pointer: 400b59  :f0 48 89 c6 48 8d 3d 64
pointer: 400b5a  :48 89 c6 48 8d 3d 64 5
pointer: 400b5b  :89 c6 48 8d 3d 64 5 0
pointer: 400b5c  :c6 48 8d 3d 64 5 0 0
pointer: 400b5d  :48 8d 3d 64 5 0 0 b8
pointer: 400b5e  :8d 3d 64 5 0 0 b8 0
pointer: 400b5f  :3d 64 5 0 0 b8 0 0
pointer: 400b60  :64 5 0 0 b8 0 0 0
pointer: 400b61  :5 0 0 b8 0 0 0 0
pointer: 400b62  :0 0 b8 0 0 0 0 e8
pointer: 400b63  :0 b8 0 0 0 0 e8 72
pointer: 400b64  :b8 0 0 0 0 e8 72 fb
pointer: 400b65  :0 0 0 0 e8 72 fb ff
pointer: 400b66  :0 0 0 e8 72 fb ff ff
pointer: 400b67  :0 0 e8 72 fb ff ff 48
pointer: 400b68  :0 e8 72 fb ff ff 48 8b
pointer: 400b69  :e8 72 fb ff ff 48 8b 45
pointer: 400b6a  :72 fb ff ff 48 8b 45 f8
pointer: 400b6b  :fb ff ff 48 8b 45 f8 48
pointer: 400b6c  :ff ff 48 8b 45 f8 48 89
pointer: 400b6d  :ff 48 8b 45 f8 48 89 c6
pointer: 400b6e  :48 8b 45 f8 48 89 c6 48
pointer: 400b6f  :8b 45 f8 48 89 c6 48 8d
pointer: 400b70  :45 f8 48 89 c6 48 8d 3d
pointer: 400b71  :f8 48 89 c6 48 8d 3d 60
```

```
pointer: 400b72  :48 89 c6 48 8d 3d 60 5
pointer: 400b73  :89 c6 48 8d 3d 60 5 0
pointer: 400b74  :c6 48 8d 3d 60 5 0 0
pointer: 400b75  :48 8d 3d 60 5 0 0 b8
pointer: 400b76  :8d 3d 60 5 0 0 b8 0
pointer: 400b77  :3d 60 5 0 0 b8 0 0
pointer: 400b78  :60 5 0 0 b8 0 0 0
pointer: 400b79  :5 0 0 b8 0 0 0 0
pointer: 400b7a  :0 0 b8 0 0 0 0 e8
pointer: 400b7b  :0 b8 0 0 0 0 e8 5a
pointer: 400b7c  :b8 0 0 0 0 e8 5a fb
pointer: 400b7d  :0 0 0 0 e8 5a fb ff
pointer: 400b7e  :0 0 0 e8 5a fb ff ff
pointer: 400b7f  :0 0 e8 5a fb ff ff 90
pointer: 400b80  :0 e8 5a fb ff ff 90 c9
pointer: 400b81  :e8 5a fb ff ff 90 c9 c3
pointer: 400b82  :5a fb ff ff 90 c9 c3 55
pointer: 400b83  :fb ff ff 90 c9 c3 55 48
pointer: 400b84  :ff ff 90 c9 c3 55 48 89
pointer: 400b85  :ff 90 c9 c3 55 48 89 e5
pointer: 400b86  :90 c9 c3 55 48 89 e5 bf
pointer: 400b87  :c9 c3 55 48 89 e5 bf 1
pointer: 400b88  :c3 55 48 89 e5 bf 1 0
pointer: 400b89  :55 48 89 e5 bf 1 0 0
pointer: 400b8a  :48 89 e5 bf 1 0 0 0
pointer: 400b8b  :89 e5 bf 1 0 0 0 b8
pointer: 400b8c  :e5 bf 1 0 0 0 b8 0
pointer: 400b8d  :bf 1 0 0 0 b8 0 0
pointer: 400b8e  :1 0 0 0 b8 0 0 0
pointer: 400b8f  :0 0 0 b8 0 0 0 0
pointer: 400b90  :0 0 b8 0 0 0 0 e8
pointer: 400b91  :0 b8 0 0 0 0 e8 b9
pointer: 400b92  :b8 0 0 0 0 e8 b9 2
pointer: 400b93  :0 0 0 0 e8 b9 2 0
```

```
pointer: 400b94  :0 0 0 e8 b9 2 0 0
pointer: 400b95  :0 0 e8 b9 2 0 0 48
pointer: 400b96  :0 e8 b9 2 0 0 48 8d
pointer: 400b97  :e8 b9 2 0 0 48 8d 3d
pointer: 400b98  :b9 2 0 0 48 8d 3d 4d
pointer: 400b99  :2 0 0 48 8d 3d 4d 5
pointer: 400b9a  :0 0 48 8d 3d 4d 5 0
pointer: 400b9b  :0 48 8d 3d 4d 5 0 0
pointer: 400b9c  :48 8d 3d 4d 5 0 0 e8
pointer: 400b9d  :8d 3d 4d 5 0 0 e8 f8
pointer: 400b9e  :3d 4d 5 0 0 e8 f8 fa
pointer: 400b9f  :4d 5 0 0 e8 f8 fa ff
pointer: 400ba0  :5 0 0 e8 f8 fa ff ff
pointer: 400ba1  :0 0 e8 f8 fa ff ff 90
pointer: 400ba2  :0 e8 f8 fa ff ff 90 5d
pointer: 400ba3  :e8 f8 fa ff ff 90 5d c3
pointer: 400ba4  :f8 fa ff ff 90 5d c3 55
pointer: 400ba5  :fa ff ff 90 5d c3 55 48
pointer: 400ba6  :ff ff 90 5d c3 55 48 89
pointer: 400ba7  :ff 90 5d c3 55 48 89 e5
pointer: 400ba8  :90 5d c3 55 48 89 e5 48
pointer: 400ba9  :5d c3 55 48 89 e5 48 83
pointer: 400baa  :c3 55 48 89 e5 48 83 ec
pointer: 400bab  :55 48 89 e5 48 83 ec 10
pointer: 400bac  :48 89 e5 48 83 ec 10 ba
pointer: 400bad  :89 e5 48 83 ec 10 ba 80
pointer: 400bae  :e5 48 83 ec 10 ba 80 1
pointer: 400baf  :48 83 ec 10 ba 80 1 0
pointer: 400bb0  :83 ec 10 ba 80 1 0 0
pointer: 400bb1  :ec 10 ba 80 1 0 0 be
pointer: 400bb2  :10 ba 80 1 0 0 be 2
pointer: 400bb3  :ba 80 1 0 0 be 2 0
pointer: 400bb4  :80 1 0 0 be 2 0 0
pointer: 400bb5  :1 0 0 be 2 0 0 0
```

```
pointer: 400bb6  :0 0 be 2 0 0 0 48
pointer: 400bb7  :0 be 2 0 0 0 48 8d
pointer: 400bb8  :be 2 0 0 0 48 8d 3d
pointer: 400bb9  :2 0 0 0 48 8d 3d 48
pointer: 400bba  :0 0 0 48 8d 3d 48 5
pointer: 400bbb  :0 0 48 8d 3d 48 5 0
pointer: 400bbc  :0 48 8d 3d 48 5 0 0
pointer: 400bbd  :48 8d 3d 48 5 0 0 b8
pointer: 400bbe  :8d 3d 48 5 0 0 b8 0
pointer: 400bbf  :3d 48 5 0 0 b8 0 0
pointer: 400bc0  :48 5 0 0 b8 0 0 0
pointer: 400bc1  :5 0 0 b8 0 0 0 0
pointer: 400bc2  :0 0 b8 0 0 0 0 e8
pointer: 400bc3  :0 b8 0 0 0 0 e8 52
pointer: 400bc4  :b8 0 0 0 0 e8 52 fb
pointer: 400bc5  :0 0 0 0 e8 52 fb ff
pointer: 400bc6  :0 0 0 e8 52 fb ff ff
pointer: 400bc7  :0 0 e8 52 fb ff ff 89
pointer: 400bc8  :0 e8 52 fb ff ff 89 45
pointer: 400bc9  :e8 52 fb ff ff 89 45 f4
pointer: 400bca  :52 fb ff ff 89 45 f4 83
pointer: 400bcb  :fb ff ff 89 45 f4 83 7d
pointer: 400bcc  :ff ff 89 45 f4 83 7d f4
pointer: 400bcd  :ff 89 45 f4 83 7d f4 0
pointer: 400bce  :89 45 f4 83 7d f4 0 79
pointer: 400bcf  :45 f4 83 7d f4 0 79 16
pointer: 400bd0  :f4 83 7d f4 0 79 16 48
pointer: 400bd1  :83 7d f4 0 79 16 48 8d
pointer: 400bd2  :7d f4 0 79 16 48 8d 3d
pointer: 400bd3  :f4 0 79 16 48 8d 3d 3a
pointer: 400bd4  :0 79 16 48 8d 3d 3a 5
pointer: 400bd5  :79 16 48 8d 3d 3a 5 0
pointer: 400bd6  :16 48 8d 3d 3a 5 0 0
pointer: 400bd7  :48 8d 3d 3a 5 0 0 e8
```

```
pointer: 400bd8  :8d 3d 3a 5 0 0 e8 4d
pointer: 400bd9  :3d 3a 5 0 0 e8 4d fb
pointer: 400bda  :3a 5 0 0 e8 4d fb ff
pointer: 400bdb  :5 0 0 e8 4d fb ff ff
pointer: 400bdc  :0 0 e8 4d fb ff ff bf
pointer: 400bdd  :0 e8 4d fb ff ff bf 1
pointer: 400bde  :e8 4d fb ff ff bf 1 0
pointer: 400bdf  :4d fb ff ff bf 1 0 0
pointer: 400be0  :fb ff ff bf 1 0 0 0
pointer: 400be1  :ff ff bf 1 0 0 0 e8
pointer: 400be2  :ff bf 1 0 0 0 e8 53
pointer: 400be3  :bf 1 0 0 0 e8 53 fb
pointer: 400be4  :1 0 0 0 e8 53 fb ff
pointer: 400be5  :0 0 0 e8 53 fb ff ff
pointer: 400be6  :0 0 e8 53 fb ff ff 8b
pointer: 400be7  :0 e8 53 fb ff ff 8b 45
pointer: 400be8  :e8 53 fb ff ff 8b 45 f4
pointer: 400be9  :53 fb ff ff 8b 45 f4 41
pointer: 400bea  :fb ff ff 8b 45 f4 41 b9
pointer: 400beb  :ff ff 8b 45 f4 41 b9 0
pointer: 400bec  :ff 8b 45 f4 41 b9 0 0
pointer: 400bed  :8b 45 f4 41 b9 0 0 0
pointer: 400bee  :45 f4 41 b9 0 0 0 0
pointer: 400bef  :f4 41 b9 0 0 0 0 41
pointer: 400bf0  :41 b9 0 0 0 0 41 89
pointer: 400bf1  :b9 0 0 0 0 41 89 c0
pointer: 400bf2  :0 0 0 0 41 89 c0 b9
pointer: 400bf3  :0 0 0 41 89 c0 b9 2
pointer: 400bf4  :0 0 41 89 c0 b9 2 0
pointer: 400bf5  :0 41 89 c0 b9 2 0 0
pointer: 400bf6  :41 89 c0 b9 2 0 0 0
pointer: 400bf7  :89 c0 b9 2 0 0 0 ba
pointer: 400bf8  :c0 b9 2 0 0 0 ba 3
pointer: 400bf9  :b9 2 0 0 0 ba 3 0
```

```
pointer: 400bfa  :2 0 0 0 ba 3 0 0
pointer: 400bfb  :0 0 0 ba 3 0 0 0
pointer: 400bfc  :0 0 ba 3 0 0 0 be
pointer: 400bfd  :0 ba 3 0 0 0 be 0
pointer: 400bfe  :ba 3 0 0 0 be 0 a3
pointer: 400bff  :3 0 0 0 be 0 a3 e1
pointer: 400c00  :0 0 0 be 0 a3 e1 11
pointer: 400c01  :0 0 be 0 a3 e1 11 bf
pointer: 400c02  :0 be 0 a3 e1 11 bf 0
pointer: 400c03  :be 0 a3 e1 11 bf 0 0
pointer: 400c04  :0 a3 e1 11 bf 0 0 0
pointer: 400c05  :a3 e1 11 bf 0 0 0 0
pointer: 400c06  :e1 11 bf 0 0 0 0 e8
pointer: 400c07  :11 bf 0 0 0 0 e8 be
pointer: 400c08  :bf 0 0 0 0 e8 be fa
pointer: 400c09  :0 0 0 0 e8 be fa ff
pointer: 400c0a  :0 0 0 e8 be fa ff ff
pointer: 400c0b  :0 0 e8 be fa ff ff 48
pointer: 400c0c  :0 e8 be fa ff ff 48 89
pointer: 400c0d  :e8 be fa ff ff 48 89 45
pointer: 400c0e  :be fa ff ff 48 89 45 f8
pointer: 400c0f  :fa ff ff 48 89 45 f8 48
pointer: 400c10  :ff ff 48 89 45 f8 48 8b
pointer: 400c11  :ff 48 89 45 f8 48 8b 45
pointer: 400c12  :48 89 45 f8 48 8b 45 f8
pointer: 400c13  :89 45 f8 48 8b 45 f8 48
pointer: 400c14  :45 f8 48 8b 45 f8 48 89
pointer: 400c15  :f8 48 8b 45 f8 48 89 5
pointer: 400c16  :48 8b 45 f8 48 89 5 3f
pointer: 400c17  :8b 45 f8 48 89 5 3f 15
pointer: 400c18  :45 f8 48 89 5 3f 15 20
pointer: 400c19  :f8 48 89 5 3f 15 20 0
pointer: 400c1a  :48 89 5 3f 15 20 0 48
pointer: 400c1b  :89 5 3f 15 20 0 48 8b
```

```
pointer: 400c1c  :5 3f 15 20 0 48 8b 5
pointer: 400c1d  :3f 15 20 0 48 8b 5 38
pointer: 400c1e  :15 20 0 48 8b 5 38 15
pointer: 400c1f  :20 0 48 8b 5 38 15 20
pointer: 400c20  :0 48 8b 5 38 15 20 0
pointer: 400c21  :48 8b 5 38 15 20 0 48
pointer: 400c22  :8b 5 38 15 20 0 48 83
pointer: 400c23  :5 38 15 20 0 48 83 f8
pointer: 400c24  :38 15 20 0 48 83 f8 ff
pointer: 400c25  :15 20 0 48 83 f8 ff 75
pointer: 400c26  :20 0 48 83 f8 ff 75 16
pointer: 400c27  :0 48 83 f8 ff 75 16 48
pointer: 400c28  :48 83 f8 ff 75 16 48 8d
pointer: 400c29  :83 f8 ff 75 16 48 8d 3d
pointer: 400c2a  :f8 ff 75 16 48 8d 3d a
pointer: 400c2b  :ff 75 16 48 8d 3d a 5
pointer: 400c2c  :75 16 48 8d 3d a 5 0
pointer: 400c2d  :16 48 8d 3d a 5 0 0
pointer: 400c2e  :48 8d 3d a 5 0 0 e8
pointer: 400c2f  :8d 3d a 5 0 0 e8 f6
pointer: 400c30  :3d a 5 0 0 e8 f6 fa
pointer: 400c31  :a 5 0 0 e8 f6 fa ff
pointer: 400c32  :5 0 0 e8 f6 fa ff ff
pointer: 400c33  :0 0 e8 f6 fa ff ff bf
pointer: 400c34  :0 e8 f6 fa ff ff bf 1
pointer: 400c35  :e8 f6 fa ff ff bf 1 0
pointer: 400c36  :f6 fa ff ff bf 1 0 0
```

OBJDUMP:

```
app:     file format elf64-x86-64
app
architecture: i386:x86-64, flags 0x00000112:
```

```
EXEC_P, HAS_SYMS, D_PAGED
start address 0x0000000000400750


Program Header:
    PHDR off    0x0000000000000040 vaddr 0x0000000000400040 paddr
0x0000000000400040 align 2**3
         filesz 0x00000000000001f8 memsz 0x00000000000001f8 flags r--
  INTERP off    0x0000000000000238 vaddr 0x0000000000400238 paddr
0x0000000000400238 align 2**0
         filesz 0x000000000000001c memsz 0x000000000000001c flags r--
    LOAD off    0x0000000000000000 vaddr 0x0000000000400000 paddr
0x0000000000400000 align 2**21
         filesz 0x00000000000014c0 memsz 0x00000000000014c0 flags r-x
    LOAD off    0x0000000000001e10 vaddr 0x0000000000601e10 paddr
0x0000000000601e10 align 2**21
         filesz 0x0000000000000278 memsz 0x00000000000041f0 flags rw-
 DYNAMIC off    0x0000000000001e20 vaddr 0x0000000000601e20 paddr
0x0000000000601e20 align 2**3
         filesz 0x00000000000001d0 memsz 0x00000000000001d0 flags rw-
    NOTE off    0x0000000000000254 vaddr 0x0000000000400254 paddr
0x0000000000400254 align 2**2
         filesz 0x0000000000000044 memsz 0x0000000000000044 flags r--
EH_FRAME off    0x00000000000011f4 vaddr 0x00000000004011f4 paddr
0x00000000004011f4 align 2**2
         filesz 0x000000000000008c memsz 0x000000000000008c flags r--
   STACK off    0x0000000000000000 vaddr 0x0000000000000000 paddr
0x0000000000000000 align 2**4
         filesz 0x0000000000000000 memsz 0x0000000000000000 flags rw-
   RELRO off    0x0000000000001e10 vaddr 0x0000000000601e10 paddr
0x0000000000601e10 align 2**0
         filesz 0x00000000000001f0 memsz 0x00000000000001f0 flags r--


Dynamic Section:
  NEEDED               libc.so.6
  INIT                 0x0000000000400660
  FINI                 0x0000000000400f34
  INIT_ARRAY           0x0000000000601e10
  INIT_ARRAYSZ        0x0000000000000008
  FINI_ARRAY          0x0000000000601e18
  FINI_ARRAYSZ        0x0000000000000008
  GNU_HASH            0x0000000000400298
  STRTAB              0x0000000000400420
  SYMTAB              0x00000000004002b8
  STRSZ               0x000000000000009b
  SYMENT              0x0000000000000018
  DEBUG               0x0000000000000000
  PLTGOT              0x0000000000602000
  PLTRELSZ            0x0000000000000120
  PLTREL              0x0000000000000007
  JMPREL              0x0000000000400540
  RELA                0x0000000000400510
  RELASZ              0x0000000000000030
  RELAENT             0x0000000000000018
  VERNEED             0x00000000004004e0
  VERNEEDNUM          0x0000000000000001
  VERSYM              0x00000000004004bc

Version References:
  required from libc.so.6:
    0x0d696914 0x00 03 GLIBC_2.4
    0x09691a75 0x00 02 GLIBC_2.2.5

Sections:
Idx Name          Size      VMA               LMA               File off  Algn
  0 .interp       0000001c  0000000000400238  0000000000400238  00000238  2**0
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  1 .note.ABI-tag 00000020  0000000000400254  0000000000400254  00000254  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  2 .note.gnu.build-id 00000024  0000000000400274  0000000000400274  00000274
2**2
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
```

```
 3 .gnu.hash     0000001c  0000000000400298  0000000000400298  00000298  2**3
                 CONTENTS, ALLOC, LOAD, READONLY, DATA
 4 .dynsym       00000168  00000000004002b8  00000000004002b8  000002b8  2**3
                 CONTENTS, ALLOC, LOAD, READONLY, DATA
 5 .dynstr       0000009b  0000000000400420  0000000000400420  00000420  2**0
                 CONTENTS, ALLOC, LOAD, READONLY, DATA
 6 .gnu.version  0000001e  00000000004004bc  00000000004004bc  000004bc  2**1
                 CONTENTS, ALLOC, LOAD, READONLY, DATA
 7 .gnu.version_r 00000030  00000000004004e0  00000000004004e0  000004e0  2**3
                 CONTENTS, ALLOC, LOAD, READONLY, DATA
 8 .rela.dyn     00000030  0000000000400510  0000000000400510  00000510  2**3
                 CONTENTS, ALLOC, LOAD, READONLY, DATA
 9 .rela.plt     00000120  0000000000400540  0000000000400540  00000540  2**3
                 CONTENTS, ALLOC, LOAD, READONLY, DATA
10 .init         00000017  0000000000400660  0000000000400660  00000660  2**2
                 CONTENTS, ALLOC, LOAD, READONLY, CODE
11 .plt          000000d0  0000000000400680  0000000000400680  00000680  2**4
                 CONTENTS, ALLOC, LOAD, READONLY, CODE
12 .text         000007e2  0000000000400750  0000000000400750  00000750  2**4
                 CONTENTS, ALLOC, LOAD, READONLY, CODE
13 .fini         00000009  0000000000400f34  0000000000400f34  00000f34  2**2
                 CONTENTS, ALLOC, LOAD, READONLY, CODE
14 .rodata       000002b4  0000000000400f40  0000000000400f40  00000f40  2**3
                 CONTENTS, ALLOC, LOAD, READONLY, DATA
15 .eh_frame_hdr 0000008c  00000000004011f4  00000000004011f4  000011f4  2**2
                 CONTENTS, ALLOC, LOAD, READONLY, DATA
16 .eh_frame     00000240  0000000000401280  0000000000401280  00001280  2**3
                 CONTENTS, ALLOC, LOAD, READONLY, DATA
17 .init_array   00000008  0000000000601e10  0000000000601e10  00001e10  2**3
                 CONTENTS, ALLOC, LOAD, DATA
18 .fini_array   00000008  0000000000601e18  0000000000601e18  00001e18  2**3
                 CONTENTS, ALLOC, LOAD, DATA
19 .dynamic      000001d0  0000000000601e20  0000000000601e20  00001e20  2**3
                 CONTENTS, ALLOC, LOAD, DATA
 20 .got          00000010  0000000000601ff0  0000000000601ff0  00001ff0  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 21 .got.plt      00000078  0000000000602000  0000000000602000  00002000  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 22 .data         00000010  0000000000602078  0000000000602078  00002078  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 23 .bss          00003f60  00000000006020a0  00000000006020a0  00002088  2**5
                  ALLOC
 24 .comment      00000029  0000000000000000  0000000000000000  00002088  2**0
                  CONTENTS, READONLY
SYMBOL TABLE:
0000000000400238 l    d  .interp        0000000000000000              .interp
0000000000400254 l    d  .note.ABI-tag  0000000000000000              .note.ABI-tag
0000000000400274 l    d  .note.gnu.build-id     0000000000000000
.note.gnu.build-id
0000000000400298 l    d  .gnu.hash      0000000000000000              .gnu.hash
00000000004002b8 l    d  .dynsym        0000000000000000              .dynsym
0000000000400420 l    d  .dynstr        0000000000000000              .dynstr
00000000004004bc l    d  .gnu.version   0000000000000000              .gnu.version
00000000004004e0 l    d  .gnu.version_r 0000000000000000              .gnu.version_r
0000000000400510 l    d  .rela.dyn      0000000000000000              .rela.dyn
0000000000400540 l    d  .rela.plt      0000000000000000              .rela.plt
0000000000400660 l    d  .init  0000000000000000              .init
0000000000400680 l    d  .plt   0000000000000000              .plt
0000000000400750 l    d  .text  0000000000000000              .text
0000000000400f34 l    d  .fini  0000000000000000              .fini
0000000000400f40 l    d  .rodata        0000000000000000              .rodata
00000000004011f4 l    d  .eh_frame_hdr  0000000000000000              .eh_frame_hdr
0000000000401280 l    d  .eh_frame      0000000000000000              .eh_frame
0000000000601e10 l    d  .init_array    0000000000000000              .init_array
0000000000601e18 l    d  .fini_array    0000000000000000              .fini_array
0000000000601e20 l    d  .dynamic       0000000000000000              .dynamic
0000000000601ff0 l    d  .got   0000000000000000              .got
0000000000602000 l    d  .got.plt       0000000000000000              .got.plt
```

```
0000000000602078 l    d  .data  0000000000000000           .data
00000000006020a0 l    d  .bss   0000000000000000           .bss
0000000000000000 l    d  .comment       0000000000000000           .comment
0000000000000000 l    df *ABS*  0000000000000000           crtstuff.c
0000000000400790 l     F .text  0000000000000000           deregister_tm_clones
00000000004007c0 l     F .text  0000000000000000           register_tm_clones
0000000000400800 l     F .text  0000000000000000           __do_global_dtors_aux
00000000006020a0 l     O .bss   0000000000000001           completed.7698
0000000000601e18 l     O .fini_array    0000000000000000
__do_global_dtors_aux_fini_array_entry
0000000000400830 l     F .text  0000000000000000           frame_dummy
0000000000601e10 l     O .init_array    0000000000000000
__frame_dummy_init_array_entry
0000000000000000 l    df *ABS*  0000000000000000           module1.c
0000000000000000 l    df *ABS*  0000000000000000           module2.c
0000000000000000 l    df *ABS*  0000000000000000           module3.c
0000000000000000 l    df *ABS*  0000000000000000           crtstuff.c
00000000004014bc l     O .eh_frame      0000000000000000           __FRAME_END__
0000000000000000 l    df *ABS*  0000000000000000
0000000000601e18 l       .init_array    0000000000000000           __init_array_end
0000000000601e20 l     O .dynamic       0000000000000000           _DYNAMIC
0000000000601e10 l       .init_array    0000000000000000           __init_array_start
00000000004011f4 l       .eh_frame_hdr  0000000000000000
__GNU_EH_FRAME_HDR
0000000000602000 l     O .got.plt       0000000000000000
_GLOBAL_OFFSET_TABLE_
0000000000400f30 g     F .text  0000000000000002           __libc_csu_fini
0000000000000000       F *UND*  0000000000000000           putchar@@GLIBC_2.2.5
0000000000400c6e g     F .text  00000000000000f3           step5
0000000000400b89 g     F .text  0000000000000022           step3
000000000040094c g     F .text  00000000000001b4           step1
0000000000602078  w      .data  0000000000000000           data_start
0000000000400e41 g     F .text  0000000000000014           add
0000000000000000       F *UND*  0000000000000000           puts@@GLIBC_2.2.5
0000000000000000       F *UND*  0000000000000000           getpid@@GLIBC_2.2.5
```

```
0000000000602088 g       .data  0000000000000000             _edata
0000000000400f34 g     F .fini  0000000000000000             _fini
0000000000400df6 g     F .text  000000000000002c             findAverage
0000000000000000       F *UND*  0000000000000000
__stack_chk_fail@@GLIBC_2.4
0000000000000000       F *UND*  0000000000000000             mmap@@GLIBC_2.2.5
0000000000000000       F *UND*  0000000000000000             printf@@GLIBC_2.2.5
0000000000602160 g     O .bss   0000000000000008             mem_buffer
0000000000602100 g     O .bss   0000000000000020             init3
00000000006020c0 g     O .bss   0000000000000020             init1
0000000000000000       F *UND*  0000000000000000             strncat@@GLIBC_2.2.5
0000000000000000       F *UND*  0000000000000000
__libc_start_main@@GLIBC_2.2.5
0000000000602078 g       .data  0000000000000000             __data_start
0000000000000000       F *UND*  0000000000000000             getchar@@GLIBC_2.2.5
0000000000602180 g     O .bss   0000000000000fa0             uninit4
0000000000603120 g     O .bss   0000000000000fa0             uninit2
0000000000000000  w      *UND*  0000000000000000             __gmon_start__
0000000000602080 g     O .data  0000000000000000             .hidden __dso_handle
0000000000602140 g     O .bss   0000000000000004             averageCount
0000000000400f40 g     O .rodata        0000000000000004             _IO_stdin_used
0000000000400bab g     F .text  00000000000000c3             step4
0000000000400b00 g     F .text  0000000000000089             step2
0000000000400ec0 g     F .text  0000000000000065             __libc_csu_init
0000000000400d61 g     F .text  0000000000000095             step6
0000000000000000       F *UND*  0000000000000000             malloc@@GLIBC_2.2.5
0000000000606000 g       .bss   0000000000000000             _end
0000000000400780 g     F .text  0000000000000002             .hidden
_dl_relocate_static_pie
0000000000400750 g     F .text  000000000000002b             _start
0000000000602088 g       .bss   0000000000000000             __bss_start
0000000000400837 g     F .text  0000000000000115             main
0000000000400e55 g     F .text  000000000000006b             recursiveFunction
0000000000400e22 g     F .text  000000000000001f             findSquare
0000000000000000       F *UND*  0000000000000000             open@@GLIBC_2.2.5
```

```
0000000000000000       F *UND*  0000000000000000             perror@@GLIBC_2.2.5
0000000000602120 g     O .bss   0000000000000020            init4
00000000006020e0 g     O .bss   0000000000000020            init2
0000000000000000       F *UND*  0000000000000000             exit@@GLIBC_2.2.5
0000000000602088 g     O .data  0000000000000000            .hidden __TMC_END__
00000000006040c0 g     O .bss   0000000000000fa0           uninit1
0000000000605060 g     O .bss   0000000000000fa0           uninit3
0000000000400660 g     F .init  0000000000000000          _init
0000000000602144 g     O .bss   0000000000000004            squareCount



Disassembly of section .init:


0000000000400660 <_init>:
  400660:       48 83 ec 08             sub    $0x8,%rsp
  400664:       48 8b 05 8d 19 20 00    mov    0x20198d(%rip),%rax        # 601ff8
<__gmon_start__>
  40066b:       48 85 c0                test   %rax,%rax
  40066e:       74 02                   je     400672 <_init+0x12>
  400670:       ff d0                   callq  *%rax
  400672:       48 83 c4 08             add    $0x8,%rsp
  400676:       c3                      retq

Disassembly of section .plt:


0000000000400680 <.plt>:
  400680:       ff 35 82 19 20 00       pushq  0x201982(%rip)        # 602008
<_GLOBAL_OFFSET_TABLE_+0x8>
  400686:       ff 25 84 19 20 00       jmpq   *0x201984(%rip)        # 602010
<_GLOBAL_OFFSET_TABLE_+0x10>
  40068c:       0f 1f 40 00             nopl   0x0(%rax)

0000000000400690 <putchar@plt>:
  400690:	ff 25 82 19 20 00    	jmpq   *0x201982(%rip)        # 602018
<putchar@GLIBC_2.2.5>
  400696:	68 00 00 00 00       	pushq  $0x0
  40069b:	e9 e0 ff ff ff       	jmpq   400680 <.plt>

00000000004006a0 <puts@plt>:
  4006a0:	ff 25 7a 19 20 00    	jmpq   *0x20197a(%rip)        # 602020
<puts@GLIBC_2.2.5>
  4006a6:	68 01 00 00 00       	pushq  $0x1
  4006ab:	e9 d0 ff ff ff       	jmpq   400680 <.plt>

00000000004006b0 <getpid@plt>:
  4006b0:	ff 25 72 19 20 00    	jmpq   *0x201972(%rip)        # 602028
<getpid@GLIBC_2.2.5>
  4006b6:	68 02 00 00 00       	pushq  $0x2
  4006bb:	e9 c0 ff ff ff       	jmpq   400680 <.plt>

00000000004006c0 <__stack_chk_fail@plt>:
  4006c0:	ff 25 6a 19 20 00    	jmpq   *0x20196a(%rip)        # 602030
<__stack_chk_fail@GLIBC_2.4>
  4006c6:	68 03 00 00 00       	pushq  $0x3
  4006cb:	e9 b0 ff ff ff       	jmpq   400680 <.plt>

00000000004006d0 <mmap@plt>:
  4006d0:	ff 25 62 19 20 00    	jmpq   *0x201962(%rip)        # 602038
<mmap@GLIBC_2.2.5>
  4006d6:	68 04 00 00 00       	pushq  $0x4
  4006db:	e9 a0 ff ff ff       	jmpq   400680 <.plt>

00000000004006e0 <printf@plt>:
  4006e0:	ff 25 5a 19 20 00    	jmpq   *0x20195a(%rip)        # 602040
<printf@GLIBC_2.2.5>
  4006e6:	68 05 00 00 00       	pushq  $0x5
  4006eb:	e9 90 ff ff ff       	jmpq   400680 <.plt>

00000000004006f0 <strncat@plt>:
  4006f0:       ff 25 52 19 20 00       jmpq   *0x201952(%rip)        # 602048
<strncat@GLIBC_2.2.5>

  4006f6:       68 06 00 00 00          pushq  $0x6

  4006fb:       e9 80 ff ff ff          jmpq   400680 <.plt>


0000000000400700 <getchar@plt>:

  400700:       ff 25 4a 19 20 00       jmpq   *0x20194a(%rip)        # 602050
<getchar@GLIBC_2.2.5>

  400706:       68 07 00 00 00          pushq  $0x7

  40070b:       e9 70 ff ff ff          jmpq   400680 <.plt>


0000000000400710 <malloc@plt>:

  400710:       ff 25 42 19 20 00       jmpq   *0x201942(%rip)        # 602058
<malloc@GLIBC_2.2.5>

  400716:       68 08 00 00 00          pushq  $0x8

  40071b:       e9 60 ff ff ff          jmpq   400680 <.plt>


0000000000400720 <open@plt>:

  400720:       ff 25 3a 19 20 00       jmpq   *0x20193a(%rip)        # 602060
<open@GLIBC_2.2.5>

  400726:       68 09 00 00 00          pushq  $0x9

  40072b:       e9 50 ff ff ff          jmpq   400680 <.plt>


0000000000400730 <perror@plt>:

  400730:       ff 25 32 19 20 00       jmpq   *0x201932(%rip)        # 602068
<perror@GLIBC_2.2.5>

  400736:       68 0a 00 00 00          pushq  $0xa

  40073b:       e9 40 ff ff ff          jmpq   400680 <.plt>


0000000000400740 <exit@plt>:

  400740:       ff 25 2a 19 20 00       jmpq   *0x20192a(%rip)        # 602070
<exit@GLIBC_2.2.5>

  400746:       68 0b 00 00 00          pushq  $0xb

  40074b:       e9 30 ff ff ff          jmpq   400680 <.plt>


Disassembly of section .text:
```

```
0000000000400750 <_start>:
  400750:       31 ed                   xor    %ebp,%ebp
  400752:       49 89 d1                mov    %rdx,%r9
  400755:       5e                      pop    %rsi
  400756:       48 89 e2                mov    %rsp,%rdx
  400759:       48 83 e4 f0             and    $0xfffffffffffffff0,%rsp
  40075d:       50                      push   %rax
  40075e:       54                      push   %rsp
  40075f:       49 c7 c0 30 0f 40 00    mov    $0x400f30,%r8
  400766:       48 c7 c1 c0 0e 40 00    mov    $0x400ec0,%rcx
  40076d:       48 c7 c7 37 08 40 00    mov    $0x400837,%rdi
  400774:       ff 15 76 18 20 00       callq  *0x201876(%rip)        # 601ff0
<__libc_start_main@GLIBC_2.2.5>
  40077a:       f4                      hlt
  40077b:       0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)

0000000000400780 <_dl_relocate_static_pie>:
  400780:       f3 c3                   repz retq
  400782:       66 2e 0f 1f 84 00 00    nopw   %cs:0x0(%rax,%rax,1)
  400789:       00 00 00
  40078c:       0f 1f 40 00             nopl   0x0(%rax)

0000000000400790 <deregister_tm_clones>:
  400790:       55                      push   %rbp
  400791:       b8 88 20 60 00          mov    $0x602088,%eax
  400796:       48 3d 88 20 60 00       cmp    $0x602088,%rax
  40079c:       48 89 e5                mov    %rsp,%rbp
  40079f:       74 17                   je     4007b8 <deregister_tm_clones+0x28>
  4007a1:       b8 00 00 00 00          mov    $0x0,%eax
  4007a6:       48 85 c0                test   %rax,%rax
  4007a9:       74 0d                   je     4007b8 <deregister_tm_clones+0x28>
  4007ab:       5d                      pop    %rbp
  4007ac:       bf 88 20 60 00          mov    $0x602088,%edi
  4007b1:       ff e0                   jmpq   *%rax
  4007b3:       0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)
  4007b8:       5d                      pop    %rbp
  4007b9:       c3                      retq
  4007ba:       66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)

00000000004007c0 <register_tm_clones>:
  4007c0:       be 88 20 60 00          mov    $0x602088,%esi
  4007c5:       55                      push   %rbp
  4007c6:       48 81 ee 88 20 60 00    sub    $0x602088,%rsi
  4007cd:       48 89 e5                mov    %rsp,%rbp
  4007d0:       48 c1 fe 03             sar    $0x3,%rsi
  4007d4:       48 89 f0                mov    %rsi,%rax
  4007d7:       48 c1 e8 3f             shr    $0x3f,%rax
  4007db:       48 01 c6                add    %rax,%rsi
  4007de:       48 d1 fe                sar    %rsi
  4007e1:       74 15                   je     4007f8 <register_tm_clones+0x38>
  4007e3:       b8 00 00 00 00          mov    $0x0,%eax
  4007e8:       48 85 c0                test   %rax,%rax
  4007eb:       74 0b                   je     4007f8 <register_tm_clones+0x38>
  4007ed:       5d                      pop    %rbp
  4007ee:       bf 88 20 60 00          mov    $0x602088,%edi
  4007f3:       ff e0                   jmpq   *%rax
  4007f5:       0f 1f 00                nopl   (%rax)
  4007f8:       5d                      pop    %rbp
  4007f9:       c3                      retq
  4007fa:       66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)

0000000000400800 <__do_global_dtors_aux>:
  400800:       80 3d 99 18 20 00 00    cmpb   $0x0,0x201899(%rip)        # 6020a0
<completed.7698>
  400807:       75 17                   jne    400820 <__do_global_dtors_aux+0x20>
  400809:       55                      push   %rbp
  40080a:       48 89 e5                mov    %rsp,%rbp
  40080d:       e8 7e ff ff ff          callq  400790 <deregister_tm_clones>
  400812:       c6 05 87 18 20 00 01    movb   $0x1,0x201887(%rip)        # 6020a0
<completed.7698>
  400819:       5d                      pop    %rbp
  40081a:       c3                      retq
  40081b:       0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)
  400820:       f3 c3                   repz retq
  400822:       0f 1f 40 00             nopl   0x0(%rax)
  400826:       66 2e 0f 1f 84 00 00    nopw   %cs:0x0(%rax,%rax,1)
  40082d:       00 00 00

0000000000400830 <frame_dummy>:
  400830:       55                      push   %rbp
  400831:       48 89 e5                mov    %rsp,%rbp
  400834:       5d                      pop    %rbp
  400835:       eb 89                   jmp    4007c0 <register_tm_clones>

0000000000400837 <main>:
  400837:       55                      push   %rbp
  400838:       48 89 e5                mov    %rsp,%rbp
  40083b:       48 83 ec 20             sub    $0x20,%rsp
  40083f:       89 7d ec                mov    %edi,-0x14(%rbp)
  400842:       48 89 75 e0             mov    %rsi,-0x20(%rbp)
  400846:       e8 65 fe ff ff          callq  4006b0 <getpid@plt>
  40084b:       89 45 fc                mov    %eax,-0x4(%rbp)
  40084e:       8b 45 fc                mov    -0x4(%rbp),%eax
  400851:       89 c6                   mov    %eax,%esi
  400853:       48 8d 3d ee 06 00 00    lea    0x6ee(%rip),%rdi        # 400f48
<_IO_stdin_used+0x8>
  40085a:       b8 00 00 00 00          mov    $0x0,%eax
  40085f:       e8 7c fe ff ff          callq  4006e0 <printf@plt>
  400864:       bf 02 00 00 00          mov    $0x2,%edi
  400869:       b8 00 00 00 00          mov    $0x0,%eax
  40086e:       e8 af 05 00 00          callq  400e22 <findSquare>
  400873:       c6 45 f7 6e             movb   $0x6e,-0x9(%rbp)
  400877:       c7 45 f8 00 00 00 00    movl   $0x0,-0x8(%rbp)
  40087e:       e9 b8 00 00 00          jmpq   40093b <main+0x104>
  400883:       83 7d f8 05             cmpl   $0x5,-0x8(%rbp)
  400887:       77 7e                   ja     400907 <main+0xd0>
  400889:       8b 45 f8                mov    -0x8(%rbp),%eax
  40088c:       48 8d 14 85 00 00 00    lea    0x0(,%rax,4),%rdx
  400893:       00
  400894:       48 8d 05 d9 06 00 00    lea    0x6d9(%rip),%rax        # 400f74
<_IO_stdin_used+0x34>
  40089b:       8b 04 02                mov    (%rdx,%rax,1),%eax
  40089e:       48 63 d0                movslq %eax,%rdx
  4008a1:       48 8d 05 cc 06 00 00    lea    0x6cc(%rip),%rax        # 400f74
<_IO_stdin_used+0x34>
  4008a8:       48 01 d0                add    %rdx,%rax
  4008ab:       ff e0                   jmpq   *%rax
  4008ad:       b8 00 00 00 00          mov    $0x0,%eax
  4008b2:       e8 95 00 00 00          callq  40094c <step1>
  4008b7:       83 45 f8 01             addl   $0x1,-0x8(%rbp)
  4008bb:       eb 60                   jmp    40091d <main+0xe6>
  4008bd:       b8 00 00 00 00          mov    $0x0,%eax
  4008c2:       e8 39 02 00 00          callq  400b00 <step2>
  4008c7:       83 45 f8 01             addl   $0x1,-0x8(%rbp)
  4008cb:       eb 50                   jmp    40091d <main+0xe6>
  4008cd:       b8 00 00 00 00          mov    $0x0,%eax
  4008d2:       e8 b2 02 00 00          callq  400b89 <step3>
  4008d7:       83 45 f8 01             addl   $0x1,-0x8(%rbp)
  4008db:       eb 40                   jmp    40091d <main+0xe6>
  4008dd:       b8 00 00 00 00          mov    $0x0,%eax
  4008e2:       e8 c4 02 00 00          callq  400bab <step4>
  4008e7:       83 45 f8 01             addl   $0x1,-0x8(%rbp)
  4008eb:       eb 30                   jmp    40091d <main+0xe6>
  4008ed:       b8 00 00 00 00          mov    $0x0,%eax
  4008f2:       e8 77 03 00 00          callq  400c6e <step5>
  4008f7:	83 45 f8 01          	addl   $0x1,-0x8(%rbp)
  4008fb:	eb 20                	jmp    40091d <main+0xe6>
  4008fd:	b8 00 00 00 00       	mov    $0x0,%eax
  400902:	e8 5a 04 00 00       	callq  400d61 <step6>
  400907:	48 8d 3d 44 06 00 00 	lea    0x644(%rip),%rdi        # 400f52
<_IO_stdin_used+0x12>
  40090e:	e8 8d fd ff ff       	callq  4006a0 <puts@plt>
  400913:	bf 01 00 00 00       	mov    $0x1,%edi
  400918:	e8 23 fe ff ff       	callq  400740 <exit@plt>
  40091d:	48 8d 3d 42 06 00 00 	lea    0x642(%rip),%rdi        # 400f66
<_IO_stdin_used+0x26>
  400924:	b8 00 00 00 00       	mov    $0x0,%eax
  400929:	e8 b2 fd ff ff       	callq  4006e0 <printf@plt>
  40092e:	e8 cd fd ff ff       	callq  400700 <getchar@plt>
  400933:	88 45 f7             	mov    %al,-0x9(%rbp)
  400936:	e8 c5 fd ff ff       	callq  400700 <getchar@plt>
  40093b:	80 7d f7 6e          	cmpb   $0x6e,-0x9(%rbp)
  40093f:	0f 84 3e ff ff ff    	je     400883 <main+0x4c>
  400945:	b8 00 00 00 00       	mov    $0x0,%eax
  40094a:	c9                   	leaveq
  40094b:	c3                   	retq

000000000040094c <step1>:
  40094c:	55                   	push   %rbp
  40094d:	48 89 e5             	mov    %rsp,%rbp
  400950:	48 8d 3d 35 06 00 00 	lea    0x635(%rip),%rdi        # 400f8c
<_IO_stdin_used+0x4c>
  400957:	e8 44 fd ff ff       	callq  4006a0 <puts@plt>
  40095c:	48 8d 05 5d 37 20 00 	lea    0x20375d(%rip),%rax        # 6040c0
<uninit1>
  400963:	48 89 c6             	mov    %rax,%rsi
  400966:	48 8d 3d 25 06 00 00 	lea    0x625(%rip),%rdi        # 400f92
<_IO_stdin_used+0x52>
  40096d:	b8 00 00 00 00       	mov    $0x0,%eax
  400972:	e8 69 fd ff ff       	callq  4006e0 <printf@plt>
  400977:       48 8d 05 a2 27 20 00    lea    0x2027a2(%rip),%rax        # 603120
<uninit2>
  40097e:       48 89 c6                mov    %rax,%rsi
  400981:       48 8d 3d 1c 06 00 00    lea    0x61c(%rip),%rdi        # 400fa4
<_IO_stdin_used+0x64>
  400988:       b8 00 00 00 00          mov    $0x0,%eax
  40098d:       e8 4e fd ff ff          callq  4006e0 <printf@plt>
  400992:       48 8d 05 c7 46 20 00    lea    0x2046c7(%rip),%rax        # 605060
<uninit3>
  400999:       48 89 c6                mov    %rax,%rsi
  40099c:       48 8d 3d 13 06 00 00    lea    0x613(%rip),%rdi        # 400fb6
<_IO_stdin_used+0x76>
  4009a3:       b8 00 00 00 00          mov    $0x0,%eax
  4009a8:       e8 33 fd ff ff          callq  4006e0 <printf@plt>
  4009ad:       48 8d 05 cc 17 20 00    lea    0x2017cc(%rip),%rax        # 602180 <uninit4>
  4009b4:       48 89 c6                mov    %rax,%rsi
  4009b7:       48 8d 3d 0a 06 00 00    lea    0x60a(%rip),%rdi        # 400fc8
<_IO_stdin_used+0x88>
  4009be:       b8 00 00 00 00          mov    $0x0,%eax
  4009c3:       e8 18 fd ff ff          callq  4006e0 <printf@plt>
  4009c8:       48 8d 05 f1 16 20 00    lea    0x2016f1(%rip),%rax        # 6020c0 <init1>
  4009cf:       48 89 c6                mov    %rax,%rsi
  4009d2:       48 8d 3d 01 06 00 00    lea    0x601(%rip),%rdi        # 400fda
<_IO_stdin_used+0x9a>
  4009d9:       b8 00 00 00 00          mov    $0x0,%eax
  4009de:       e8 fd fc ff ff          callq  4006e0 <printf@plt>
  4009e3:       48 8d 05 f6 16 20 00    lea    0x2016f6(%rip),%rax        # 6020e0 <init2>
  4009ea:       48 89 c6                mov    %rax,%rsi
  4009ed:       48 8d 3d f6 05 00 00    lea    0x5f6(%rip),%rdi        # 400fea
<_IO_stdin_used+0xaa>
  4009f4:       b8 00 00 00 00          mov    $0x0,%eax
  4009f9:       e8 e2 fc ff ff          callq  4006e0 <printf@plt>
  4009fe:       48 8d 05 fb 16 20 00    lea    0x2016fb(%rip),%rax        # 602100 <init3>
  400a05:       48 89 c6                mov    %rax,%rsi
  400a08:       48 8d 3d eb 05 00 00    lea    0x5eb(%rip),%rdi        # 400ffa
<_IO_stdin_used+0xba>
  400a0f:       b8 00 00 00 00          mov    $0x0,%eax
  400a14:	e8 c7 fc ff ff       	callq  4006e0 <printf@plt>
  400a19:	48 8d 05 00 17 20 00 	lea    0x201700(%rip),%rax        # 602120 <init4>
  400a20:	48 89 c6             	mov    %rax,%rsi
  400a23:	48 8d 3d e0 05 00 00 	lea    0x5e0(%rip),%rdi        # 40100a <_IO_stdin_used+0xca>
  400a2a:	b8 00 00 00 00       	mov    $0x0,%eax
  400a2f:	e8 ac fc ff ff       	callq  4006e0 <printf@plt>
  400a34:	48 8d 3d df 05 00 00 	lea    0x5df(%rip),%rdi        # 40101a <_IO_stdin_used+0xda>
  400a3b:	e8 60 fc ff ff       	callq  4006a0 <puts@plt>
  400a40:	48 8d 05 f0 fd ff ff 	lea    -0x210(%rip),%rax        # 400837 <main>
  400a47:	48 89 c6             	mov    %rax,%rsi
  400a4a:	48 8d 3d d4 05 00 00 	lea    0x5d4(%rip),%rdi        # 401025 <_IO_stdin_used+0xe5>
  400a51:	b8 00 00 00 00       	mov    $0x0,%eax
  400a56:	e8 85 fc ff ff       	callq  4006e0 <printf@plt>
  400a5b:	48 8d 05 ea fe ff ff 	lea    -0x116(%rip),%rax        # 40094c <step1>
  400a62:	48 89 c6             	mov    %rax,%rsi
  400a65:	48 8d 3d ca 05 00 00 	lea    0x5ca(%rip),%rdi        # 401036 <_IO_stdin_used+0xf6>
  400a6c:	b8 00 00 00 00       	mov    $0x0,%eax
  400a71:	e8 6a fc ff ff       	callq  4006e0 <printf@plt>
  400a76:	48 8d 05 83 00 00 00 	lea    0x83(%rip),%rax        # 400b00 <step2>
  400a7d:	48 89 c6             	mov    %rax,%rsi
  400a80:	48 8d 3d c1 05 00 00 	lea    0x5c1(%rip),%rdi        # 401048 <_IO_stdin_used+0x108>
  400a87:	b8 00 00 00 00       	mov    $0x0,%eax
  400a8c:	e8 4f fc ff ff       	callq  4006e0 <printf@plt>
  400a91:	48 8d 05 f1 00 00 00 	lea    0xf1(%rip),%rax        # 400b89 <step3>
  400a98:	48 89 c6             	mov    %rax,%rsi
  400a9b:	48 8d 3d b8 05 00 00 	lea    0x5b8(%rip),%rdi        # 40105a <_IO_stdin_used+0x11a>
  400aa2:	b8 00 00 00 00       	mov    $0x0,%eax
  400aa7:	e8 34 fc ff ff       	callq  4006e0 <printf@plt>
  400aac:	48 8d 05 f8 00 00 00 	lea    0xf8(%rip),%rax        # 400bab <step4>
  400ab3:	48 89 c6             	mov    %rax,%rsi
```

```
 400ab6:	48 8d 3d af 05 00 00 	lea    0x5af(%rip),%rdi        # 40106c
<_IO_stdin_used+0x12c>
 400abd:	b8 00 00 00 00       	mov    $0x0,%eax
 400ac2:	e8 19 fc ff ff       	callq  4006e0 <printf@plt>
 400ac7:	48 8d 05 a0 01 00 00 	lea    0x1a0(%rip),%rax        # 400c6e <step5>
 400ace:	48 89 c6             	mov    %rax,%rsi
 400ad1:	48 8d 3d a6 05 00 00 	lea    0x5a6(%rip),%rdi        # 40107e
<_IO_stdin_used+0x13e>
 400ad8:	b8 00 00 00 00       	mov    $0x0,%eax
 400add:	e8 fe fb ff ff       	callq  4006e0 <printf@plt>
 400ae2:	48 8d 05 78 02 00 00 	lea    0x278(%rip),%rax        # 400d61 <step6>
 400ae9:	48 89 c6             	mov    %rax,%rsi
 400aec:	48 8d 3d 9d 05 00 00 	lea    0x59d(%rip),%rdi        # 401090
<_IO_stdin_used+0x150>
 400af3:	b8 00 00 00 00       	mov    $0x0,%eax
 400af8:	e8 e3 fb ff ff       	callq  4006e0 <printf@plt>
 400afd:	90                   	nop
 400afe:	5d                   	pop    %rbp
 400aff:	c3                   	retq

0000000000400b00 <step2>:
 400b00:	55                   	push   %rbp
 400b01:	48 89 e5             	mov    %rsp,%rbp
 400b04:	48 83 ec 20          	sub    $0x20,%rsp
 400b08:	bf 80 84 1e 00       	mov    $0x1e8480,%edi
 400b0d:	e8 fe fb ff ff       	callq  400710 <malloc@plt>
 400b12:	48 89 45 e8          	mov    %rax,-0x18(%rbp)
 400b16:	bf 40 42 0f 00       	mov    $0xf4240,%edi
 400b1b:	e8 f0 fb ff ff       	callq  400710 <malloc@plt>
 400b20:	48 89 45 f0          	mov    %rax,-0x10(%rbp)
 400b24:	bf 40 42 0f 00       	mov    $0xf4240,%edi
 400b29:	e8 e2 fb ff ff       	callq  400710 <malloc@plt>
 400b2e:	48 89 45 f8          	mov    %rax,-0x8(%rbp)
 400b32:	48 8d 3d 69 05 00 00 	lea    0x569(%rip),%rdi        # 4010a2
<_IO_stdin_used+0x162>
```

```
  400b39:	e8 62 fb ff ff       	callq  4006a0 <puts@plt>
  400b3e:	48 8b 45 e8          	mov    -0x18(%rbp),%rax
  400b42:	48 89 c6             	mov    %rax,%rsi
  400b45:	48 8d 3d 68 05 00 00 	lea    0x568(%rip),%rdi        # 4010b4 <_IO_stdin_used+0x174>
  400b4c:	b8 00 00 00 00       	mov    $0x0,%eax
  400b51:	e8 8a fb ff ff       	callq  4006e0 <printf@plt>
  400b56:	48 8b 45 f0          	mov    -0x10(%rbp),%rax
  400b5a:	48 89 c6             	mov    %rax,%rsi
  400b5d:	48 8d 3d 64 05 00 00 	lea    0x564(%rip),%rdi        # 4010c8 <_IO_stdin_used+0x188>
  400b64:	b8 00 00 00 00       	mov    $0x0,%eax
  400b69:	e8 72 fb ff ff       	callq  4006e0 <printf@plt>
  400b6e:	48 8b 45 f8          	mov    -0x8(%rbp),%rax
  400b72:	48 89 c6             	mov    %rax,%rsi
  400b75:	48 8d 3d 60 05 00 00 	lea    0x560(%rip),%rdi        # 4010dc <_IO_stdin_used+0x19c>
  400b7c:	b8 00 00 00 00       	mov    $0x0,%eax
  400b81:	e8 5a fb ff ff       	callq  4006e0 <printf@plt>
  400b86:	90                   	nop
  400b87:	c9                   	leaveq
  400b88:	c3                   	retq

0000000000400b89 <step3>:
  400b89:	55                   	push   %rbp
  400b8a:	48 89 e5             	mov    %rsp,%rbp
  400b8d:	bf 01 00 00 00       	mov    $0x1,%edi
  400b92:	b8 00 00 00 00       	mov    $0x0,%eax
  400b97:	e8 b9 02 00 00       	callq  400e55 <recursiveFunction>
  400b9c:	48 8d 3d 4d 05 00 00 	lea    0x54d(%rip),%rdi        # 4010f0 <_IO_stdin_used+0x1b0>
  400ba3:	e8 f8 fa ff ff       	callq  4006a0 <puts@plt>
  400ba8:	90                   	nop
  400ba9:	5d                   	pop    %rbp
  400baa:	c3                   	retq
```

```
0000000000400bab <step4>:
  400bab:       55                      push   %rbp
  400bac:       48 89 e5                mov    %rsp,%rbp
  400baf:       48 83 ec 10             sub    $0x10,%rsp
  400bb3:       ba 80 01 00 00          mov    $0x180,%edx
  400bb8:       be 02 00 00 00          mov    $0x2,%esi
  400bbd:       48 8d 3d 48 05 00 00    lea    0x548(%rip),%rdi        # 40110c
<_IO_stdin_used+0x1cc>
  400bc4:       b8 00 00 00 00          mov    $0x0,%eax
  400bc9:       e8 52 fb ff ff          callq  400720 <open@plt>
  400bce:       89 45 f4                mov    %eax,-0xc(%rbp)
  400bd1:       83 7d f4 00             cmpl   $0x0,-0xc(%rbp)
  400bd5:       79 16                   jns    400bed <step4+0x42>
  400bd7:       48 8d 3d 3a 05 00 00    lea    0x53a(%rip),%rdi        # 401118
<_IO_stdin_used+0x1d8>
  400bde:       e8 4d fb ff ff          callq  400730 <perror@plt>
  400be3:       bf 01 00 00 00          mov    $0x1,%edi
  400be8:       e8 53 fb ff ff          callq  400740 <exit@plt>
  400bed:       8b 45 f4                mov    -0xc(%rbp),%eax
  400bf0:       41 b9 00 00 00 00       mov    $0x0,%r9d
  400bf6:       41 89 c0                mov    %eax,%r8d
  400bf9:       b9 02 00 00 00          mov    $0x2,%ecx
  400bfe:       ba 03 00 00 00          mov    $0x3,%edx
  400c03:       be 00 a3 e1 11          mov    $0x11e1a300,%esi
  400c08:       bf 00 00 00 00          mov    $0x0,%edi
  400c0d:       e8 be fa ff ff          callq  4006d0 <mmap@plt>
  400c12:       48 89 45 f8             mov    %rax,-0x8(%rbp)
  400c16:       48 8b 45 f8             mov    -0x8(%rbp),%rax
  400c1a:       48 89 05 3f 15 20 00    mov    %rax,0x20153f(%rip)        # 602160
<mem_buffer>
  400c21:       48 8b 05 38 15 20 00    mov    0x201538(%rip),%rax        # 602160
<mem_buffer>
  400c28:       48 83 f8 ff             cmp    $0xffffffffffffffff,%rax
  400c2c:       75 16                   jne    400c44 <step4+0x99>
  400c2e:       48 8d 3d 0a 05 00 00    lea    0x50a(%rip),%rdi        # 40113f
<_IO_stdin_used+0x1ff>
  400c35:       e8 f6 fa ff ff          callq  400730 <perror@plt>
  400c3a:       bf 01 00 00 00          mov    $0x1,%edi
  400c3f:       e8 fc fa ff ff          callq  400740 <exit@plt>
  400c44:       48 8d 3d 0e 05 00 00    lea    0x50e(%rip),%rdi        # 401159
<_IO_stdin_used+0x219>
  400c4b:       e8 50 fa ff ff          callq  4006a0 <puts@plt>
  400c50:       48 8b 05 09 15 20 00    mov    0x201509(%rip),%rax        # 602160
<mem_buffer>
  400c57:       48 89 c6                mov    %rax,%rsi
  400c5a:       48 8d 3d 12 05 00 00    lea    0x512(%rip),%rdi        # 401173
<_IO_stdin_used+0x233>
  400c61:       b8 00 00 00 00          mov    $0x0,%eax
  400c66:       e8 75 fa ff ff          callq  4006e0 <printf@plt>
  400c6b:       90                      nop
  400c6c:       c9                      leaveq
  400c6d:       c3                      retq

0000000000400c6e <step5>:
  400c6e:       55                      push   %rbp
  400c6f:       48 89 e5                mov    %rsp,%rbp
  400c72:       48 81 ec 70 c3 00 00    sub    $0xc370,%rsp
  400c79:       64 48 8b 04 25 28 00    mov    %fs:0x28,%rax
  400c80:       00 00
  400c82:       48 89 45 f8             mov    %rax,-0x8(%rbp)
  400c86:       31 c0                   xor    %eax,%eax
  400c88:       c7 85 94 3c ff ff 00    movl   $0x0,-0xc36c(%rbp)
  400c8f:       00 00 00
  400c92:       eb 40                   jmp    400cd4 <step5+0x66>
  400c94:       48 8b 15 c5 14 20 00    mov    0x2014c5(%rip),%rdx        # 602160
<mem_buffer>
  400c9b:       8b 85 94 3c ff ff       mov    -0xc36c(%rbp),%eax
  400ca1:       48 98                   cltq
  400ca3:       48 01 d0                add    %rdx,%rax
  400ca6:       0f b6 00                movzbl (%rax),%eax
  400ca9:       88 85 93 3c ff ff       mov    %al,-0xc36d(%rbp)
  400caf:       48 8d 8d 93 3c ff ff    lea    -0xc36d(%rbp),%rcx
  400cb6:       48 8d 85 a0 3c ff ff    lea    -0xc360(%rbp),%rax
  400cbd:       ba 01 00 00 00          mov    $0x1,%edx
  400cc2:       48 89 ce                mov    %rcx,%rsi
  400cc5:       48 89 c7                mov    %rax,%rdi
  400cc8:       e8 23 fa ff ff          callq  4006f0 <strncat@plt>
  400ccd:       83 85 94 3c ff ff 01    addl   $0x1,-0xc36c(%rbp)
  400cd4:       81 bd 94 3c ff ff f3    cmpl   $0x1f3,-0xc36c(%rbp)
  400cdb:       01 00 00
  400cde:       7e b4                   jle    400c94 <step5+0x26>
  400ce0:       48 8d 85 a0 3c ff ff    lea    -0xc360(%rbp),%rax
  400ce7:       48 89 c6                mov    %rax,%rsi
  400cea:       48 8d 3d 98 04 00 00    lea    0x498(%rip),%rdi        # 401189
<_IO_stdin_used+0x249>
  400cf1:       b8 00 00 00 00          mov    $0x0,%eax
  400cf6:       e8 e5 f9 ff ff          callq  4006e0 <printf@plt>
  400cfb:       c7 85 98 3c ff ff f4    movl   $0x1f4,-0xc368(%rbp)
  400d02:       01 00 00
  400d05:       eb 2b                   jmp    400d32 <step5+0xc4>
  400d07:       c7 85 9c 3c ff ff 6f    movl   $0x6f,-0xc364(%rbp)
  400d0e:       00 00 00
  400d11:       48 8b 15 48 14 20 00    mov    0x201448(%rip),%rdx        # 602160
<mem_buffer>
  400d18:       8b 85 98 3c ff ff       mov    -0xc368(%rbp),%eax
  400d1e:       48 98                   cltq
  400d20:       48 01 d0                add    %rdx,%rax
  400d23:       8b 95 9c 3c ff ff       mov    -0xc364(%rbp),%edx
  400d29:       88 10                   mov    %dl,(%rax)
  400d2b:       83 85 98 3c ff ff 01    addl   $0x1,-0xc368(%rbp)
  400d32:       81 bd 98 3c ff ff e7    cmpl   $0x3e7,-0xc368(%rbp)
  400d39:       03 00 00
  400d3c:       7e c9                   jle    400d07 <step5+0x99>
  400d3e:       48 8d 3d 5b 04 00 00    lea    0x45b(%rip),%rdi        # 4011a0
<_IO_stdin_used+0x260>
```

```
  400d45:	e8 56 f9 ff ff       	callq  4006a0 <puts@plt>
  400d4a:	90                   	nop
  400d4b:	48 8b 45 f8          	mov    -0x8(%rbp),%rax
  400d4f:	64 48 33 04 25 28 00 	xor    %fs:0x28,%rax
  400d56:	00 00 
  400d58:	74 05                	je     400d5f <step5+0xf1>
  400d5a:	e8 61 f9 ff ff       	callq  4006c0 <__stack_chk_fail@plt>
  400d5f:	c9                   	leaveq 
  400d60:	c3                   	retq   

0000000000400d61 <step6>:
  400d61:	55                   	push   %rbp
  400d62:	48 89 e5             	mov    %rsp,%rbp
  400d65:	48 83 ec 10          	sub    $0x10,%rsp
  400d69:	48 8d 05 c7 fa ff ff 	lea    -0x539(%rip),%rax        # 400837 <main>
  400d70:	48 89 45 f8          	mov    %rax,-0x8(%rbp)
  400d74:	c7 45 f0 00 00 00 00 	movl   $0x0,-0x10(%rbp)
  400d7b:	eb 6d                	jmp    400dea <step6+0x89>
  400d7d:	8b 45 f0             	mov    -0x10(%rbp),%eax
  400d80:	48 63 d0             	movslq %eax,%rdx
  400d83:	48 8b 45 f8          	mov    -0x8(%rbp),%rax
  400d87:	48 01 d0             	add    %rdx,%rax
  400d8a:	48 89 c6             	mov    %rax,%rsi
  400d8d:	48 8d 3d 3a 04 00 00 	lea    0x43a(%rip),%rdi        # 4011ce
<_IO_stdin_used+0x28e>
  400d94:	b8 00 00 00 00       	mov    $0x0,%eax
  400d99:	e8 42 f9 ff ff       	callq  4006e0 <printf@plt>
  400d9e:	c7 45 f4 00 00 00 00 	movl   $0x0,-0xc(%rbp)
  400da5:	eb 2f                	jmp    400dd6 <step6+0x75>
  400da7:	8b 55 f0             	mov    -0x10(%rbp),%edx
  400daa:	8b 45 f4             	mov    -0xc(%rbp),%eax
  400dad:	01 d0                	add    %edx,%eax
  400daf:	48 63 d0             	movslq %eax,%rdx
  400db2:	48 8b 45 f8          	mov    -0x8(%rbp),%rax
  400db6:	48 01 d0             	add    %rdx,%rax
  400db9:	0f b6 00             	movzbl (%rax),%eax
  400dbc:	0f b6 c0             	movzbl %al,%eax
  400dbf:	89 c6                	mov    %eax,%esi
  400dc1:	48 8d 3d 16 04 00 00 	lea    0x416(%rip),%rdi        # 4011de
<_IO_stdin_used+0x29e>
  400dc8:	b8 00 00 00 00       	mov    $0x0,%eax
  400dcd:	e8 0e f9 ff ff       	callq  4006e0 <printf@plt>
  400dd2:	83 45 f4 01          	addl   $0x1,-0xc(%rbp)
  400dd6:	83 7d f4 07          	cmpl   $0x7,-0xc(%rbp)
  400dda:	7e cb                	jle    400da7 <step6+0x46>
  400ddc:	bf 0a 00 00 00       	mov    $0xa,%edi
  400de1:	e8 aa f8 ff ff       	callq  400690 <putchar@plt>
  400de6:	83 45 f0 01          	addl   $0x1,-0x10(%rbp)
  400dea:	81 7d f0 ff 03 00 00 	cmpl   $0x3ff,-0x10(%rbp)
  400df1:	7e 8a                	jle    400d7d <step6+0x1c>
  400df3:	90                   	nop
  400df4:	c9                   	leaveq
  400df5:	c3                   	retq

0000000000400df6 <findAverage>:
  400df6:	55                   	push   %rbp
  400df7:	48 89 e5             	mov    %rsp,%rbp
  400dfa:	89 7d fc             	mov    %edi,-0x4(%rbp)
  400dfd:	89 75 f8             	mov    %esi,-0x8(%rbp)
  400e00:	8b 05 3a 13 20 00    	mov    0x20133a(%rip),%eax        # 602140
<averageCount>
  400e06:	83 c0 01             	add    $0x1,%eax
  400e09:	89 05 31 13 20 00    	mov    %eax,0x201331(%rip)        # 602140
<averageCount>
  400e0f:	8b 55 fc             	mov    -0x4(%rbp),%edx
  400e12:	8b 45 f8             	mov    -0x8(%rbp),%eax
  400e15:	01 d0                	add    %edx,%eax
  400e17:	89 c2                	mov    %eax,%edx
  400e19:	c1 ea 1f             	shr    $0x1f,%edx
  400e1c:	01 d0                	add    %edx,%eax
  400e1e:	d1 f8                	sar    %eax
  400e20:	5d                   	pop    %rbp
  400e21:	c3                   	retq

0000000000400e22 <findSquare>:
  400e22:	55                   	push   %rbp
  400e23:	48 89 e5             	mov    %rsp,%rbp
  400e26:	89 7d fc             	mov    %edi,-0x4(%rbp)
  400e29:	8b 05 15 13 20 00    	mov    0x201315(%rip),%eax        # 602144 <squareCount>
  400e2f:	83 c0 01             	add    $0x1,%eax
  400e32:	89 05 0c 13 20 00    	mov    %eax,0x20130c(%rip)        # 602144 <squareCount>
  400e38:	8b 45 fc             	mov    -0x4(%rbp),%eax
  400e3b:	0f af 45 fc          	imul   -0x4(%rbp),%eax
  400e3f:	5d                   	pop    %rbp
  400e40:	c3                   	retq

0000000000400e41 <add>:
  400e41:	55                   	push   %rbp
  400e42:	48 89 e5             	mov    %rsp,%rbp
  400e45:	89 7d fc             	mov    %edi,-0x4(%rbp)
  400e48:	89 75 f8             	mov    %esi,-0x8(%rbp)
  400e4b:	8b 55 fc             	mov    -0x4(%rbp),%edx
  400e4e:	8b 45 f8             	mov    -0x8(%rbp),%eax
  400e51:	01 d0                	add    %edx,%eax
  400e53:	5d                   	pop    %rbp
  400e54:	c3                   	retq

0000000000400e55 <recursiveFunction>:
  400e55:	55                   	push   %rbp
  400e56:	48 89 e5             	mov    %rsp,%rbp
  400e59:	48 83 ec 10          	sub    $0x10,%rsp
  400e5d:	89 7d fc             	mov    %edi,-0x4(%rbp)
  400e60:	8b 4d fc             	mov    -0x4(%rbp),%ecx
  400e63:	ba d3 4d 62 10       	mov    $0x10624dd3,%edx
  400e68:	89 c8                	mov    %ecx,%eax
  400e6a:	f7 ea                	imul   %edx
  400e6c:	c1 fa 06             	sar    $0x6,%edx
  400e6f:	89 c8                	mov    %ecx,%eax
  400e71:	c1 f8 1f             	sar    $0x1f,%eax
  400e74:	29 c2                	sub    %eax,%edx
  400e76:	89 d0                	mov    %edx,%eax
  400e78:	69 c0 e8 03 00 00    	imul   $0x3e8,%eax,%eax
  400e7e:	29 c1                	sub    %eax,%ecx
  400e80:	89 c8                	mov    %ecx,%eax
  400e82:	85 c0                	test   %eax,%eax
  400e84:	75 18                	jne    400e9e <recursiveFunction+0x49>
  400e86:	48 8d 45 fc          	lea    -0x4(%rbp),%rax
  400e8a:	48 89 c6             	mov    %rax,%rsi
  400e8d:	48 8d 3d 4e 03 00 00 	lea    0x34e(%rip),%rdi        # 4011e2
<_IO_stdin_used+0x2a2>
  400e94:	b8 00 00 00 00       	mov    $0x0,%eax
  400e99:	e8 42 f8 ff ff       	callq  4006e0 <printf@plt>
  400e9e:	8b 45 fc             	mov    -0x4(%rbp),%eax
  400ea1:	3d 50 46 00 00       	cmp    $0x4650,%eax
  400ea6:	7f 15                	jg     400ebd <recursiveFunction+0x68>
  400ea8:	8b 45 fc             	mov    -0x4(%rbp),%eax
  400eab:	83 c0 01             	add    $0x1,%eax
  400eae:	89 45 fc             	mov    %eax,-0x4(%rbp)
  400eb1:	8b 45 fc             	mov    -0x4(%rbp),%eax
  400eb4:	89 c7                	mov    %eax,%edi
  400eb6:	e8 9a ff ff ff       	callq  400e55 <recursiveFunction>
  400ebb:	eb 01                	jmp    400ebe <recursiveFunction+0x69>
  400ebd:	90                   	nop
  400ebe:	c9                   	leaveq
  400ebf:	c3                   	retq
```

```
0000000000400ec0 <__libc_csu_init>:
  400ec0:       41 57                   push   %r15
  400ec2:       41 56                   push   %r14
  400ec4:       49 89 d7                mov    %rdx,%r15
  400ec7:       41 55                   push   %r13
  400ec9:       41 54                   push   %r12
  400ecb:       4c 8d 25 3e 0f 20 00    lea    0x200f3e(%rip),%r12        # 601e10
<__frame_dummy_init_array_entry>
  400ed2:       55                      push   %rbp
  400ed3:       48 8d 2d 3e 0f 20 00    lea    0x200f3e(%rip),%rbp        # 601e18
<__init_array_end>
  400eda:       53                      push   %rbx
  400edb:       41 89 fd                mov    %edi,%r13d
  400ede:       49 89 f6                mov    %rsi,%r14
  400ee1:       4c 29 e5                sub    %r12,%rbp
  400ee4:       48 83 ec 08             sub    $0x8,%rsp
  400ee8:       48 c1 fd 03             sar    $0x3,%rbp
  400eec:       e8 6f f7 ff ff          callq  400660 <_init>
  400ef1:       48 85 ed                test   %rbp,%rbp
  400ef4:       74 20                   je     400f16 <__libc_csu_init+0x56>
  400ef6:       31 db                   xor    %ebx,%ebx
  400ef8:       0f 1f 84 00 00 00 00    nopl   0x0(%rax,%rax,1)
  400eff:       00
  400f00:       4c 89 fa                mov    %r15,%rdx
  400f03:       4c 89 f6                mov    %r14,%rsi
  400f06:       44 89 ef                mov    %r13d,%edi
  400f09:       41 ff 14 dc             callq  *(%r12,%rbx,8)
  400f0d:       48 83 c3 01             add    $0x1,%rbx
  400f11:       48 39 dd                cmp    %rbx,%rbp
  400f14:       75 ea                   jne    400f00 <__libc_csu_init+0x40>
  400f16:       48 83 c4 08             add    $0x8,%rsp
  400f1a:       5b                      pop    %rbx
  400f1b:       5d                      pop    %rbp
```

```
 400f1c:      41 5c                 pop    %r12
 400f1e:      41 5d                 pop    %r13
 400f20:      41 5e                 pop    %r14
 400f22:      41 5f                 pop    %r15
 400f24:      c3                    retq
 400f25:      90                    nop
 400f26:      66 2e 0f 1f 84 00 00    nopw   %cs:0x0(%rax,%rax,1)
 400f2d:      00 00 00

0000000000400f30 <__libc_csu_fini>:
 400f30:      f3 c3                 repz retq

Disassembly of section .fini:

0000000000400f34 <_fini>:
 400f34:      48 83 ec 08           sub    $0x8,%rsp
 400f38:      48 83 c4 08           add    $0x8,%rsp
 400f3c:      c3                    retq
```

STATIC COMPILATION:

The file size increased from 13.6KB to 866.1KB on disk after compilation using –static tag

STATIC DUMP:

```
495c26:      48 85 f6              test   %rsi,%rsi
 495c29:      74 2d                 je     495c58 <free_mem+0x348>
 495c2b:      48 83 7a 18 00         cmpq   $0x0,0x18(%rdx)
 495c30:      0f 85 a5 fd ff ff      jne    4959db <free_mem+0xcb>
 495c36:      48 8d 4a 28            lea    0x28(%rdx),%rcx
 495c3a:      31 c0                 xor    %eax,%eax
 495c3c:      eb 11                 jmp    495c4f <free_mem+0x33f>
 495c3e:      66 90                 xchg   %ax,%ax
 495c40:      48 83 c1 10           add    $0x10,%rcx
```

```
495c44:     48 83 79 f0 00          cmpq   $0x0,-0x10(%rcx)
495c49:     0f 85 8c fd ff ff       jne    4959db <free_mem+0xcb>
495c4f:     48 83 c0 01             add    $0x1,%rax
495c53:     48 39 f0                cmp    %rsi,%rax
495c56:     75 e8                   jne    495c40 <free_mem+0x330>
495c58:     48 89 d7                mov    %rdx,%rdi
495c5b:     e8 90 af f8 ff          callq  420bf0 <__free>
495c60:     49 c7 47 08 00 00 00    movq   $0x0,0x8(%r15)
495c67:     00
495c68:     4d 8b 7e 08             mov    0x8(%r14),%r15
495c6c:     e9 3c ff ff ff          jmpq   495bad <free_mem+0x29d>
495c71:     48 8d 78 08             lea    0x8(%rax),%rdi
495c75:     48 89 54 24 08          mov    %rdx,0x8(%rsp)
495c7a:     e8 b1 fb ff ff          callq  495830 <free_slotinfo>
495c7f:     84 c0                   test   %al,%al
495c81:     0f 84 54 fd ff ff       je     4959db <free_mem+0xcb>
495c87:     48 8b 54 24 08          mov    0x8(%rsp),%rdx
495c8c:     31 c0                   xor    %eax,%eax
495c8e:     48 8b 7a 08             mov    0x8(%rdx),%rdi
495c92:     48 8b 0f                mov    (%rdi),%rcx
495c95:     48 39 c8                cmp    %rcx,%rax
495c98:     74 19                   je     495cb3 <free_mem+0x3a3>
495c9a:     48 89 c6                mov    %rax,%rsi
495c9d:     48 c1 e6 04             shl    $0x4,%rsi
495ca1:     48 83 7c 37 18 00       cmpq   $0x0,0x18(%rdi,%rsi,1)
495ca7:     0f 85 2e fd ff ff       jne    4959db <free_mem+0xcb>
495cad:     48 83 c0 01             add    $0x1,%rax
495cb1:     eb e2                   jmp    495c95 <free_mem+0x385>
495cb3:     48 89 54 24 08          mov    %rdx,0x8(%rsp)
495cb8:     e8 33 af f8 ff          callq  420bf0 <__free>
495cbd:     48 8b 54 24 08          mov    0x8(%rsp),%rdx
495cc2:     48 c7 42 08 00 00 00    movq   $0x0,0x8(%rdx)
495cc9:     00
495cca:     49 8b 57 08             mov    0x8(%r15),%rdx
```

```
  495cce:	e9 50 ff ff ff       	jmpq   495c23 <free_mem+0x313>

Disassembly of section __libc_thread_freeres_fn:

0000000000495ce0 <arena_thread_freeres>:
  495ce0:	41 57                	push   %r15
  495ce2:	41 56                	push   %r14
  495ce4:	41 55                	push   %r13
  495ce6:	41 54                	push   %r12
  495ce8:	55                   	push   %rbp
  495ce9:	53                   	push   %rbx
  495cea:	48 83 ec 68          	sub    $0x68,%rsp
  495cee:	64 4c 8b 2c 25 c8 ff 	mov    %fs:0xffffffffffffffc8,%r13
  495cf5:	ff ff 
  495cf7:	64 48 8b 04 25 28 00 	mov    %fs:0x28,%rax
  495cfe:	00 00 
  495d00:	48 89 44 24 58       	mov    %rax,0x58(%rsp)
  495d05:	31 c0                	xor    %eax,%eax
  495d07:	4d 85 ed             	test   %r13,%r13
  495d0a:	0f 84 58 01 00 00    	je     495e68 <arena_thread_freeres+0x188>
  495d10:	49 8d bd 40 02 00 00 	lea    0x240(%r13),%rdi
  495d17:	48 8d 4c 24 57       	lea    0x57(%rsp),%rcx
  495d1c:	48 8b 05 d5 b0 22 00 	mov    0x22b0d5(%rip),%rax        # 6c0df8
<__free_hook>
  495d23:	49 8d 5d 40          	lea    0x40(%r13),%rbx
  495d27:	4c 8d 25 52 8b 22 00 	lea    0x228b52(%rip),%r12        # 6be880
<main_arena>
  495d2e:	64 48 c7 04 25 c8 ff 	movq   $0x0,%fs:0xffffffffffffffc8
  495d35:	ff ff 00 00 00 00 
  495d3b:	64 c6 04 25 d0 ff ff 	movb   $0x1,%fs:0xffffffffffffffd0
  495d42:	ff 01 
  495d44:	48 89 7c 24 08       	mov    %rdi,0x8(%rsp)
  495d49:	48 89 4c 24 18       	mov    %rcx,0x18(%rsp)
  495d4e:	4c 89 6c 24 10       	mov    %r13,0x10(%rsp)
  495d53:       4c 8b 3b                mov    (%rbx),%r15
  495d56:       41 bd 01 00 00 00       mov    $0x1,%r13d
  495d5c:       4d 85 ff                test   %r15,%r15
  495d5f:       0f 84 99 00 00 00       je     495dfe <arena_thread_freeres+0x11e>
  495d65:       49 8b 17                mov    (%r15),%rdx
  495d68:       48 85 c0                test   %rax,%rax
  495d6b:       48 89 13                mov    %rdx,(%rbx)
  495d6e:       0f 85 24 03 00 00       jne    496098 <arena_thread_freeres+0x3b8>
  495d74:       49 8b 47 f8             mov    -0x8(%r15),%rax
  495d78:       4d 8d 47 f0             lea    -0x10(%r15),%r8
  495d7c:       a8 02                   test   $0x2,%al
  495d7e:       0f 84 d4 01 00 00       je     495f58 <arena_thread_freeres+0x278>
  495d84:       8b 3d 8a 8a 22 00       mov    0x228a8a(%rip),%edi        # 6be814
<mp_+0x34>
  495d8a:       48 89 c2                mov    %rax,%rdx
  495d8d:       48 83 e2 f8             and    $0xfffffffffffffff8,%rdx
  495d91:       85 ff                   test   %edi,%edi
  495d93:       75 15                   jne    495daa <arena_thread_freeres+0xca>
  495d95:       48 3b 05 54 8a 22 00    cmp    0x228a54(%rip),%rax        # 6be7f0
<mp_+0x10>
  495d9c:       76 0c                   jbe    495daa <arena_thread_freeres+0xca>
  495d9e:       48 3d 00 00 00 02       cmp    $0x2000000,%rax
  495da4:       0f 86 be 04 00 00       jbe    496268 <arena_thread_freeres+0x588>
  495daa:       49 8b 77 f0             mov    -0x10(%r15),%rsi
  495dae:       4c 89 c7                mov    %r8,%rdi
  495db1:       48 8b 05 40 94 22 00    mov    0x229440(%rip),%rax        # 6bf1f8
<_dl_pagesize>
  495db8:       48 29 f7                sub    %rsi,%rdi
  495dbb:       48 01 d6                add    %rdx,%rsi
  495dbe:       48 83 e8 01             sub    $0x1,%rax
  495dc2:       48 89 fa                mov    %rdi,%rdx
  495dc5:       48 09 f2                or     %rsi,%rdx
  495dc8:       48 85 d0                test   %rdx,%rax
  495dcb:       0f 85 af 06 00 00       jne    496480 <arena_thread_freeres+0x7a0>
  495dd1:       f0 ff 0d 30 8a 22 00    lock decl 0x228a30(%rip)        # 6be808
<mp_+0x28>
  495dd8:       48 89 f0                mov    %rsi,%rax
  495ddb:       48 f7 d8                neg    %rax
  495dde:       f0 48 01 05 32 8a 22    lock add %rax,0x228a32(%rip)        # 6be818
<mp_+0x38>
  495de5:       00
  495de6:       e8 75 92 fb ff          callq  44f060 <__munmap>
  495deb:       4c 8b 3b                mov    (%rbx),%r15
  495dee:       48 8b 05 03 b0 22 00    mov    0x22b003(%rip),%rax        # 6c0df8
<__free_hook>
  495df5:       4d 85 ff                test   %r15,%r15
  495df8:       0f 85 67 ff ff ff       jne    495d65 <arena_thread_freeres+0x85>
  495dfe:       48 83 c3 08             add    $0x8,%rbx
  495e02:       48 39 5c 24 08          cmp    %rbx,0x8(%rsp)
  495e07:       0f 85 46 ff ff ff       jne    495d53 <arena_thread_freeres+0x73>
  495e0d:       48 85 c0                test   %rax,%rax
  495e10:       4c 8b 6c 24 10          mov    0x10(%rsp),%r13
  495e15:       0f 85 1d 0b 00 00       jne    496938 <arena_thread_freeres+0xc58>
  495e1b:       49 8b 45 f8             mov    -0x8(%r13),%rax
  495e1f:       49 8d 5d f0             lea    -0x10(%r13),%rbx
  495e23:       a8 02                   test   $0x2,%al
  495e25:       0f 84 88 0a 00 00       je     4968b3 <arena_thread_freeres+0xbd3>
  495e2b:       8b 15 e3 89 22 00       mov    0x2289e3(%rip),%edx        # 6be814
<mp_+0x34>
  495e31:       85 d2                   test   %edx,%edx
  495e33:       75 28                   jne    495e5d <arena_thread_freeres+0x17d>
  495e35:       48 3b 05 b4 89 22 00    cmp    0x2289b4(%rip),%rax        # 6be7f0
<mp_+0x10>
  495e3c:       76 1f                   jbe    495e5d <arena_thread_freeres+0x17d>
  495e3e:       48 3d 00 00 00 02       cmp    $0x2000000,%rax
  495e44:       77 17                   ja     495e5d <arena_thread_freeres+0x17d>
  495e46:       48 83 e0 f8             and    $0xfffffffffffffff8,%rax
  495e4a:       48 8d 14 00             lea    (%rax,%rax,1),%rdx
  495e4e:       48 89 05 9b 89 22 00    mov    %rax,0x22899b(%rip)        # 6be7f0
<mp_+0x10>
```

```
  495e55:       48 89 15 84 89 22 00    mov    %rdx,0x228984(%rip)        # 6be7e0 <mp_>
  495e5c:       90                      nop
  495e5d:       48 89 df                mov    %rbx,%rdi
  495e60:       e8 0b 56 f8 ff          callq  41b470 <munmap_chunk>
  495e65:       0f 1f 00                nopl   (%rax)
  495e68:       48 c7 c0 d8 ff ff ff    mov    $0xffffffffffffffd8,%rax
  495e6f:       64 48 8b 10             mov    %fs:(%rax),%rdx
  495e73:       64 48 c7 00 00 00 00    movq   $0x0,%fs:(%rax)
  495e7a:       00
  495e7b:       48 85 d2                test   %rdx,%rdx
  495e7e:       0f 84 aa 00 00 00       je     495f2e <arena_thread_freeres+0x24e>
  495e84:       be 01 00 00 00          mov    $0x1,%esi
  495e89:       31 c0                   xor    %eax,%eax
  495e8b:       83 3d ba f8 22 00 00    cmpl   $0x0,0x22f8ba(%rip)        # 6c574c
<__libc_multiple_threads>
  495e92:       74 0c                   je     495ea0 <arena_thread_freeres+0x1c0>
  495e94:       f0 0f b1 35 94 af 22    lock cmpxchg %esi,0x22af94(%rip)        # 6c0e30
<free_list_lock>
  495e9b:       00
  495e9c:       75 0b                   jne    495ea9 <arena_thread_freeres+0x1c9>
  495e9e:       eb 23                   jmp    495ec3 <arena_thread_freeres+0x1e3>
  495ea0:       0f b1 35 89 af 22 00    cmpxchg %esi,0x22af89(%rip)        # 6c0e30
<free_list_lock>
  495ea7:       74 1a                   je     495ec3 <arena_thread_freeres+0x1e3>
  495ea9:       48 8d 3d 80 af 22 00    lea    0x22af80(%rip),%rdi        # 6c0e30
<free_list_lock>
  495eb0:       48 81 ec 80 00 00 00    sub    $0x80,%rsp
  495eb7:       e8 44 a9 fb ff          callq  450800 <__lll_lock_wait_private>
  495ebc:       48 81 c4 80 00 00 00    add    $0x80,%rsp
  495ec3:       48 8b 82 80 08 00 00    mov    0x880(%rdx),%rax
  495eca:       48 85 c0                test   %rax,%rax
  495ecd:       0f 84 1d 04 00 00       je     4962f0 <arena_thread_freeres+0x610>
  495ed3:       48 83 e8 01             sub    $0x1,%rax
  495ed7:       48 85 c0                test   %rax,%rax
  495eda:       48 89 82 80 08 00 00    mov    %rax,0x880(%rdx)
```

```
 495ee1:    75 15                  jne    495ef8 <arena_thread_freeres+0x218>
 495ee3:    48 8b 05 3e af 22 00    mov    0x22af3e(%rip),%rax        # 6c0e28
<free_list>
 495eea:    48 89 15 37 af 22 00    mov    %rdx,0x22af37(%rip)        # 6c0e28
<free_list>
 495ef1:    48 89 82 78 08 00 00    mov    %rax,0x878(%rdx)
 495ef8:    83 3d 4d f8 22 00 00    cmpl   $0x0,0x22f84d(%rip)        # 6c574c
<__libc_multiple_threads>
 495eff:    74 0b                  je     495f0c <arena_thread_freeres+0x22c>
 495f01:    f0 ff 0d 28 af 22 00    lock decl 0x22af28(%rip)        # 6c0e30
<free_list_lock>
 495f08:    75 0a                  jne    495f14 <arena_thread_freeres+0x234>
 495f0a:    eb 22                  jmp    495f2e <arena_thread_freeres+0x24e>
 495f0c:    ff 0d 1e af 22 00       decl   0x22af1e(%rip)        # 6c0e30 <free_list_lock>
 495f12:    74 1a                  je     495f2e <arena_thread_freeres+0x24e>
 495f14:    48 8d 3d 15 af 22 00    lea    0x22af15(%rip),%rdi        # 6c0e30
<free_list_lock>
 495f1b:    48 81 ec 80 00 00 00    sub    $0x80,%rsp
 495f22:    e8 09 a9 fb ff          callq  450830 <__lll_unlock_wake_private>
 495f27:    48 81 c4 80 00 00 00    add    $0x80,%rsp
 495f2e:    48 8b 44 24 58          mov    0x58(%rsp),%rax
 495f33:    64 48 33 04 25 28 00    xor    %fs:0x28,%rax
 495f3a:    00 00
 495f3c:    0f 85 9b 0c 00 00       jne    496bdd <arena_thread_freeres+0xefd>
 495f42:    48 83 c4 68             add    $0x68,%rsp
 495f46:    5b                     pop    %rbx
 495f47:    5d                     pop    %rbp
 495f48:    41 5c                  pop    %r12
 495f4a:    41 5d                  pop    %r13
 495f4c:    41 5e                  pop    %r14
 495f4e:    41 5f                  pop    %r15
 495f50:    c3                     retq
 495f51:    0f 1f 80 00 00 00 00    nopl   0x0(%rax)
 495f58:    64 48 83 3c 25 c8 ff    cmpq   $0x0,%fs:0xffffffffffffffc8
 495f5f:    ff ff 00
 495f62:    0f 84 f8 03 00 00       je     496360 <arena_thread_freeres+0x680>
```

```
  495f68:       a8 04                   test   $0x4,%al
  495f6a:       4c 89 e5                mov    %r12,%rbp
  495f6d:       0f 85 8d 00 00 00       jne    496000 <arena_thread_freeres+0x320>
  495f73:       49 89 c1                mov    %rax,%r9
  495f76:       49 83 e1 f8             and    $0xfffffffffffffff8,%r9
  495f7a:       4c 89 ca                mov    %r9,%rdx
  495f7d:       48 f7 da                neg    %rdx
  495f80:       4c 39 c2                cmp    %r8,%rdx
  495f83:       0f 82 b7 03 00 00       jb     496340 <arena_thread_freeres+0x660>
  495f89:       41 f6 c0 0f             test   $0xf,%r8b
  495f8d:       0f 85 ad 03 00 00       jne    496340 <arena_thread_freeres+0x660>
  495f93:       49 83 f9 1f             cmp    $0x1f,%r9
  495f97:       0f 86 b3 03 00 00       jbe    496350 <arena_thread_freeres+0x670>
  495f9d:       a8 08                   test   $0x8,%al
  495f9f:       0f 85 ab 03 00 00       jne    496350 <arena_thread_freeres+0x670>
  495fa5:       64 48 8b 0c 25 c8 ff    mov    %fs:0xffffffffffffffc8,%rcx
  495fac:       ff ff
  495fae:       48 85 c9                test   %rcx,%rcx
  495fb1:       74 65                   je     496018 <arena_thread_freeres+0x338>
  495fb3:       49 8d 51 ef             lea    -0x11(%r9),%rdx
  495fb7:       48 c1 ea 04             shr    $0x4,%rdx
  495fbb:       48 3b 15 6e 88 22 00    cmp    0x22886e(%rip),%rdx        # 6be830
<mp_+0x50>
  495fc2:       73 54                   jae    496018 <arena_thread_freeres+0x338>
  495fc4:       48 0f be 3c 11          movsbq (%rcx,%rdx,1),%rdi
  495fc9:       48 3b 3d 70 88 22 00    cmp    0x228870(%rip),%rdi        # 6be840
<mp_+0x60>
  495fd0:       48 89 fe                mov    %rdi,%rsi
  495fd3:       73 43                   jae    496018 <arena_thread_freeres+0x338>
  495fd5:       48 83 fa 3f             cmp    $0x3f,%rdx
  495fd9:       0f 87 b1 04 00 00       ja     496490 <arena_thread_freeres+0x7b0>
  495fdf:       48 8d 04 d1             lea    (%rcx,%rdx,8),%rax
  495fe3:       83 c6 01                add    $0x1,%esi
  495fe6:       48 8b 78 40             mov    0x40(%rax),%rdi
  495fea:       49 89 3f                mov    %rdi,(%r15)
  495fed:       4c 89 78 40             mov    %r15,0x40(%rax)
  495ff1:       40 88 34 11             mov    %sil,(%rcx,%rdx,1)
  495ff5:       e9 f1 fd ff ff          jmpq   495deb <arena_thread_freeres+0x10b>
  495ffa:       66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)
  496000:       4c 89 c2                mov    %r8,%rdx
  496003:       48 81 e2 00 00 00 fc    and    $0xfffffffffc000000,%rdx
  49600a:       48 8b 2a                mov    (%rdx),%rbp
  49600d:       e9 61 ff ff ff          jmpq   495f73 <arena_thread_freeres+0x293>
  496012:       66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)
  496018:       4c 3b 0d 19 ae 22 00    cmp    0x22ae19(%rip),%r9        # 6c0e38
<global_max_fast>
  49601f:       0f 87 8b 00 00 00       ja     4960b0 <arena_thread_freeres+0x3d0>
  496025:       4b 8d 14 08             lea    (%r8,%r9,1),%rdx
  496029:       48 8b 42 08             mov    0x8(%rdx),%rax
  49602d:       48 83 f8 10             cmp    $0x10,%rax
  496031:       0f 86 b1 03 00 00       jbe    4963e8 <arena_thread_freeres+0x708>
  496037:       48 83 e0 f8             and    $0xfffffffffffffff8,%rax
  49603b:       48 3b 85 88 08 00 00    cmp    0x888(%rbp),%rax
  496042:       0f 83 a0 03 00 00       jae    4963e8 <arena_thread_freeres+0x708>
  496048:       8b 05 e6 ad 22 00       mov    0x22ade6(%rip),%eax        # 6c0e34
<perturb_byte>
  49604e:       85 c0                   test   %eax,%eax
  496050:       0f 85 67 05 00 00       jne    4965bd <arena_thread_freeres+0x8dd>
  496056:       41 c1 e9 04             shr    $0x4,%r9d
  49605a:       c7 45 08 01 00 00 00    movl   $0x1,0x8(%rbp)
  496061:       41 8d 49 fe             lea    -0x2(%r9),%ecx
  496065:       8b 35 e1 f6 22 00       mov    0x22f6e1(%rip),%esi        # 6c574c
<__libc_multiple_threads>
  49606b:       48 8d 44 cd 00          lea    0x0(%rbp,%rcx,8),%rax
  496070:       85 f6                   test   %esi,%esi
  496072:       48 8b 50 10             mov    0x10(%rax),%rdx
  496076:       0f 85 50 04 00 00       jne    4964cc <arena_thread_freeres+0x7ec>
  49607c:       49 39 d0                cmp    %rdx,%r8
  49607f:       0f 84 a2 04 00 00       je     496527 <arena_thread_freeres+0x847>
  496085:       49 89 17                mov    %rdx,(%r15)
  496088:       4c 89 40 10             mov    %r8,0x10(%rax)
  49608c:       e9 5a fd ff ff          jmpq   495deb <arena_thread_freeres+0x10b>
  496091:       0f 1f 80 00 00 00 00    nopl   0x0(%rax)
  496098:       48 8b b4 24 98 00 00    mov    0x98(%rsp),%rsi
  49609f:       00
  4960a0:       4c 89 ff                mov    %r15,%rdi
  4960a3:       ff d0                   callq  *%rax
  4960a5:       e9 41 fd ff ff          jmpq   495deb <arena_thread_freeres+0x10b>
  4960aa:       66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)
  4960b0:       a8 02                   test   $0x2,%al
  4960b2:       0f 85 78 02 00 00       jne    496330 <arena_thread_freeres+0x650>
  4960b8:       8b 0d 8e f6 22 00       mov    0x22f68e(%rip),%ecx        # 6c574c
<__libc_multiple_threads>
  4960be:       41 be 01 00 00 00       mov    $0x1,%r14d
  4960c4:       85 c9                   test   %ecx,%ecx
  4960c6:       0f 85 2d 05 00 00       jne    4965f9 <arena_thread_freeres+0x919>
  4960cc:       48 8b 45 60             mov    0x60(%rbp),%rax
  4960d0:       4b 8d 0c 08             lea    (%r8,%r9,1),%rcx
  4960d4:       49 39 c0                cmp    %rax,%r8
  4960d7:       0f 84 58 05 00 00       je     496635 <arena_thread_freeres+0x955>
  4960dd:       f6 45 04 02             testb  $0x2,0x4(%rbp)
  4960e1:       0f 84 5a 05 00 00       je     496641 <arena_thread_freeres+0x961>
  4960e7:       48 8b 41 08             mov    0x8(%rcx),%rax
  4960eb:       a8 01                   test   $0x1,%al
  4960ed:       0f 84 7a 05 00 00       je     49666d <arena_thread_freeres+0x98d>
  4960f3:       49 89 c2                mov    %rax,%r10
  4960f6:       49 83 e2 f8             and    $0xfffffffffffffff8,%r10
  4960fa:       48 83 f8 10             cmp    $0x10,%rax
  4960fe:       0f 86 ad 04 00 00       jbe    4965b1 <arena_thread_freeres+0x8d1>
  496104:       4c 3b 95 88 08 00 00    cmp    0x888(%rbp),%r10
  49610b:       0f 83 a0 04 00 00       jae    4965b1 <arena_thread_freeres+0x8d1>
  496111:       8b 35 1d ad 22 00       mov    0x22ad1d(%rip),%esi        # 6c0e34
<perturb_byte>
```

```
496117:   85 f6                   test   %esi,%esi
496119:   0f 85 5a 05 00 00       jne    496679 <arena_thread_freeres+0x999>
49611f:   41 f6 47 f8 01          testb  $0x1,-0x8(%r15)
496124:   0f 85 86 00 00 00       jne    4961b0 <arena_thread_freeres+0x4d0>
49612a:   49 8b 47 f0             mov    -0x10(%r15),%rax
49612e:   49 29 c0                sub    %rax,%r8
496131:   49 01 c1                add    %rax,%r9
496134:   49 8b 70 08             mov    0x8(%r8),%rsi
496138:   48 89 f0                mov    %rsi,%rax
49613b:   48 83 e0 f8             and    $0xfffffffffffffff8,%rax
49613f:   49 3b 04 00             cmp    (%r8,%rax,1),%rax
496143:   0f 85 34 07 00 00       jne    49687d <arena_thread_freeres+0xb9d>
496149:   49 8b 40 10             mov    0x10(%r8),%rax
49614d:   49 8b 50 18             mov    0x18(%r8),%rdx
496151:   4c 3b 40 18             cmp    0x18(%rax),%r8
496155:   0f 85 06 05 00 00       jne    496661 <arena_thread_freeres+0x981>
49615b:   4c 3b 42 10             cmp    0x10(%rdx),%r8
49615f:   0f 85 fc 04 00 00       jne    496661 <arena_thread_freeres+0x981>
496165:   48 81 fe ff 03 00 00    cmp    $0x3ff,%rsi
49616c:   48 89 50 18             mov    %rdx,0x18(%rax)
496170:   48 89 42 10             mov    %rax,0x10(%rdx)
496174:   76 3a                   jbe    4961b0 <arena_thread_freeres+0x4d0>
496176:   49 8b 50 20             mov    0x20(%r8),%rdx
49617a:   48 85 d2                test   %rdx,%rdx
49617d:   74 31                   je     4961b0 <arena_thread_freeres+0x4d0>
49617f:   4c 3b 42 28             cmp    0x28(%rdx),%r8
496183:   0f 85 a2 08 00 00       jne    496a2b <arena_thread_freeres+0xd4b>
496189:   49 8b 70 28             mov    0x28(%r8),%rsi
49618d:   4c 3b 46 20             cmp    0x20(%rsi),%r8
496191:   0f 85 94 08 00 00       jne    496a2b <arena_thread_freeres+0xd4b>
496197:   48 83 78 20 00          cmpq   $0x0,0x20(%rax)
49619c:   0f 84 f3 09 00 00       je     496b95 <arena_thread_freeres+0xeb5>
4961a2:   48 89 72 28             mov    %rsi,0x28(%rdx)
4961a6:   49 8b 40 28             mov    0x28(%r8),%rax
```

```
 4961aa:	48 89 50 20          	mov    %rdx,0x20(%rax)
 4961ae:	66 90                	xchg   %ax,%ax
 4961b0:	48 3b 4d 60          	cmp    0x60(%rbp),%rcx
 4961b4:	0f 84 79 03 00 00    	je     496533 <arena_thread_freeres+0x853>
 4961ba:	42 f6 44 11 08 01    	testb  $0x1,0x8(%rcx,%r10,1)
 4961c0:	48 8b 41 08          	mov    0x8(%rcx),%rax
 4961c4:	0f 84 c6 01 00 00    	je     496390 <arena_thread_freeres+0x6b0>
 4961ca:	48 83 e0 fe          	and    $0xfffffffffffffffe,%rax
 4961ce:	48 89 41 08          	mov    %rax,0x8(%rcx)
 4961d2:	48 8b 45 70          	mov    0x70(%rbp),%rax
 4961d6:	48 8d 55 60          	lea    0x60(%rbp),%rdx
 4961da:	48 3b 50 18          	cmp    0x18(%rax),%rdx
 4961de:	0f 85 05 07 00 00    	jne    4968e9 <arena_thread_freeres+0xc09>
 4961e4:	49 81 f9 ff 03 00 00 	cmp    $0x3ff,%r9
 4961eb:	49 89 40 10          	mov    %rax,0x10(%r8)
 4961ef:	49 89 50 18          	mov    %rdx,0x18(%r8)
 4961f3:	76 10                	jbe    496205 <arena_thread_freeres+0x525>
 4961f5:	49 c7 40 20 00 00 00 	movq   $0x0,0x20(%r8)
 4961fc:	00 
 4961fd:	49 c7 40 28 00 00 00 	movq   $0x0,0x28(%r8)
 496204:	00 
 496205:	4c 89 45 70          	mov    %r8,0x70(%rbp)
 496209:	4c 89 40 18          	mov    %r8,0x18(%rax)
 49620d:	4c 89 c8             	mov    %r9,%rax
 496210:	48 83 c8 01          	or     $0x1,%rax
 496214:	49 89 40 08          	mov    %rax,0x8(%r8)
 496218:	4f 89 0c 08          	mov    %r9,(%r8,%r9,1)
 49621c:	49 81 f9 ff ff 00 00 	cmp    $0xffff,%r9
 496223:	0f 87 21 03 00 00    	ja     49654a <arena_thread_freeres+0x86a>
 496229:	45 85 f6             	test   %r14d,%r14d
 49622c:	0f 85 b9 fb ff ff    	jne    495deb <arena_thread_freeres+0x10b>
 496232:	83 3d 13 f5 22 00 00 	cmpl   $0x0,0x22f513(%rip)        # 6c574c
<__libc_multiple_threads>
 496239:	74 08                	je     496243 <arena_thread_freeres+0x563>
```

```
  49623b:       f0 ff 4d 00             lock decl 0x0(%rbp)
  49623f:       75 07                   jne    496248 <arena_thread_freeres+0x568>
  496241:       eb 1c                   jmp    49625f <arena_thread_freeres+0x57f>
  496243:       ff 4d 00                decl   0x0(%rbp)
  496246:       74 17                   je     49625f <arena_thread_freeres+0x57f>
  496248:       48 8d 7d 00             lea    0x0(%rbp),%rdi
  49624c:       48 81 ec 80 00 00 00    sub    $0x80,%rsp
  496253:       e8 d8 a5 fb ff          callq  450830 <__lll_unlock_wake_private>
  496258:       48 81 c4 80 00 00 00    add    $0x80,%rsp
  49625f:       e9 87 fb ff ff          jmpq   495deb <arena_thread_freeres+0x10b>
  496264:       0f 1f 40 00             nopl   0x0(%rax)
  496268:       48 8d 04 12             lea    (%rdx,%rdx,1),%rax
  49626c:       48 89 15 7d 85 22 00    mov    %rdx,0x22857d(%rip)        # 6be7f0
<mp_+0x10>
  496273:       48 89 05 66 85 22 00    mov    %rax,0x228566(%rip)        # 6be7e0 <mp_>
  49627a:       90                      nop
  49627b:       49 8b 57 f8             mov    -0x8(%r15),%rdx
  49627f:       f6 c2 02                test   $0x2,%dl
  496282:       0f 85 7d 09 00 00       jne    496c05 <arena_thread_freeres+0xf25>
  496288:       48 8b 15 d9 8e 22 00    mov    0x228ed9(%rip),%rdx        # 6bf168
<__progname>
  49628f:       48 8d 05 b7 0e 00 00    lea    0xeb7(%rip),%rax        # 49714d
<__PRETTY_FUNCTION__.10913+0x5d>
  496296:       48 8d 0d d8 4c 01 00    lea    0x14cd8(%rip),%rcx        # 4aaf75
<conversion_rate+0x11d>
  49629d:       48 8d 35 cf 16 00 00    lea    0x16cf(%rip),%rsi        # 497973
<__PRETTY_FUNCTION__.10520+0xb3>
  4962a4:       41 b9 0f 0b 00 00       mov    $0xb0f,%r9d
  4962aa:       80 3a 00                cmpb   $0x0,(%rdx)
  4962ad:       48 0f 45 c8             cmovne %rax,%rcx
  4962b1:       48 83 ec 08             sub    $0x8,%rsp
  4962b5:       56                      push   %rsi
  4962b6:       50                      push   %rax
  4962b7:       48 8d 05 d2 25 00 00    lea    0x25d2(%rip),%rax        # 498890
<__PRETTY_FUNCTION__.12012>
  4962be:       50                      push   %rax
```

```
4962bf:       4c 8d 05 65 16 00 00    lea    0x1665(%rip),%r8        # 49792b
<__PRETTY_FUNCTION__.10520+0x6b>
4962c6:       48 8d 35 b3 1c 00 00    lea    0x1cb3(%rip),%rsi        # 497f80
<__PRETTY_FUNCTION__.12394+0x375>
4962cd:       31 ff                   xor    %edi,%edi
4962cf:       31 c0                   xor    %eax,%eax
4962d1:       e8 7a 9e f7 ff          callq  410150 <__fxprintf>
4962d6:       48 8b 3d bb 84 22 00    mov    0x2284bb(%rip),%rdi        # 6be798
<_IO_stderr>
4962dd:       48 83 c4 20             add    $0x20,%rsp
4962e1:       e8 0a a7 f7 ff          callq  4109f0 <_IO_fflush>
4962e6:       e8 55 7f f7 ff          callq  40e240 <abort>
4962eb:       0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)
4962f0:       48 8b 15 71 8e 22 00    mov    0x228e71(%rip),%rdx        # 6bf168
<__progname>
4962f7:       48 8d 05 4f 0e 00 00    lea    0xe4f(%rip),%rax        # 49714d
<__PRETTY_FUNCTION__.10913+0x5d>
4962fe:       48 8d 0d 70 4c 01 00    lea    0x14c70(%rip),%rcx        # 4aaf75
<conversion_rate+0x11d>
496305:       48 8d 35 3f 18 00 00    lea    0x183f(%rip),%rsi        # 497b4b
<__PRETTY_FUNCTION__.10520+0x28b>
49630c:       41 b9 c0 03 00 00       mov    $0x3c0,%r9d
496312:       4c 8d 05 1b 16 00 00    lea    0x161b(%rip),%r8        # 497934
<__PRETTY_FUNCTION__.10520+0x74>
496319:       80 3a 00                cmpb   $0x0,(%rdx)
49631c:       48 0f 45 c8             cmovne %rax,%rcx
496320:       48 83 ec 08             sub    $0x8,%rsp
496324:       56                      push   %rsi
496325:       50                      push   %rax
496326:       48 8d 05 33 26 00 00    lea    0x2633(%rip),%rax        # 498960
<__PRETTY_FUNCTION__.11819>
49632d:       50                      push   %rax
49632e:       eb 96                   jmp    4962c6 <arena_thread_freeres+0x5e6>
496330:       4c 89 c7                mov    %r8,%rdi
496333:       e8 38 51 f8 ff          callq  41b470 <munmap_chunk>
496338:       e9 ae fa ff ff          jmpq   495deb <arena_thread_freeres+0x10b>
49633d:       0f 1f 00                nopl   (%rax)
```

```
  496340:       48 8d 3d 74 16 00 00    lea    0x1674(%rip),%rdi        # 4979bb
<__PRETTY_FUNCTION__.10520+0xfb>
  496347:       e8 24 3a f8 ff          callq  419d70 <malloc_printerr>
  49634c:       0f 1f 40 00             nopl   0x0(%rax)
  496350:       48 8d 3d 7c 16 00 00    lea    0x167c(%rip),%rdi        # 4979d3
<__PRETTY_FUNCTION__.10520+0x113>
  496357:       e8 14 3a f8 ff          callq  419d70 <malloc_printerr>
  49635c:       0f 1f 40 00             nopl   0x0(%rax)
  496360:       64 80 3c 25 d0 ff ff    cmpb   $0x0,%fs:0xffffffffffffffd0
  496367:       ff 00
  496369:       0f 85 f9 fb ff ff       jne    495f68 <arena_thread_freeres+0x288>
  49636f:       4c 89 44 24 20          mov    %r8,0x20(%rsp)
  496374:       e8 e7 93 f8 ff          callq  41f760 <tcache_init.part.4>
  496379:       49 8b 47 f8             mov    -0x8(%r15),%rax
  49637d:       4c 8b 44 24 20          mov    0x20(%rsp),%r8
  496382:       e9 e1 fb ff ff          jmpq   495f68 <arena_thread_freeres+0x288>
  496387:       66 0f 1f 84 00 00 00    nopw   0x0(%rax,%rax,1)
  49638e:       00 00
  496390:       48 89 c2                mov    %rax,%rdx
  496393:       48 83 e2 f8             and    $0xfffffffffffffff8,%rdx
  496397:       48 3b 14 11             cmp    (%rcx,%rdx,1),%rdx
  49639b:       0f 85 dc 04 00 00       jne    49687d <arena_thread_freeres+0xb9d>
  4963a1:       48 8b 51 10             mov    0x10(%rcx),%rdx
  4963a5:       48 8b 71 18             mov    0x18(%rcx),%rsi
  4963a9:       48 3b 4a 18             cmp    0x18(%rdx),%rcx
  4963ad:       0f 85 ae 02 00 00       jne    496661 <arena_thread_freeres+0x981>
  4963b3:       48 3b 4e 10             cmp    0x10(%rsi),%rcx
  4963b7:       0f 85 a4 02 00 00       jne    496661 <arena_thread_freeres+0x981>
  4963bd:       48 3d ff 03 00 00       cmp    $0x3ff,%rax
  4963c3:       48 89 72 18             mov    %rsi,0x18(%rdx)
  4963c7:       48 89 56 10             mov    %rdx,0x10(%rsi)
  4963cb:       76 0d                   jbe    4963da <arena_thread_freeres+0x6fa>
  4963cd:       48 8b 41 20             mov    0x20(%rcx),%rax
  4963d1:       48 85 c0                test   %rax,%rax
  4963d4:      0f 85 b3 05 00 00       jne    49698d <arena_thread_freeres+0xcad>
  4963da:      4d 01 d1                add    %r10,%r9
  4963dd:      e9 f0 fd ff ff          jmpq   4961d2 <arena_thread_freeres+0x4f2>
  4963e2:      66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)
  4963e8:      44 89 ee                mov    %r13d,%esi
  4963eb:      31 c0                   xor    %eax,%eax
  4963ed:      83 3d 58 f3 22 00 00    cmpl   $0x0,0x22f358(%rip)        # 6c574c
<__libc_multiple_threads>
  4963f4:      74 09                   je     4963ff <arena_thread_freeres+0x71f>
  4963f6:      f0 0f b1 75 00          lock cmpxchg %esi,0x0(%rbp)
  4963fb:      75 08                   jne    496405 <arena_thread_freeres+0x725>
  4963fd:      eb 1d                   jmp    49641c <arena_thread_freeres+0x73c>
  4963ff:      0f b1 75 00             cmpxchg %esi,0x0(%rbp)
  496403:      74 17                   je     49641c <arena_thread_freeres+0x73c>
  496405:      48 8d 7d 00             lea    0x0(%rbp),%rdi
  496409:      48 81 ec 80 00 00 00    sub    $0x80,%rsp
  496410:      e8 eb a3 fb ff          callq  450800 <__lll_lock_wait_private>
  496415:      48 81 c4 80 00 00 00    add    $0x80,%rsp
  49641c:      48 8b 42 08             mov    0x8(%rdx),%rax
  496420:      ba 01 00 00 00          mov    $0x1,%edx
  496425:      48 83 f8 10             cmp    $0x10,%rax
  496429:      76 10                   jbe    49643b <arena_thread_freeres+0x75b>
  49642b:      31 d2                   xor    %edx,%edx
  49642d:      48 83 e0 f8             and    $0xfffffffffffffff8,%rax
  496431:      48 3b 85 88 08 00 00    cmp    0x888(%rbp),%rax
  496438:      0f 93 c2                setae  %dl
  49643b:      83 3d 0a f3 22 00 00    cmpl   $0x0,0x22f30a(%rip)        # 6c574c
<__libc_multiple_threads>
  496442:      74 08                   je     49644c <arena_thread_freeres+0x76c>
  496444:      f0 ff 4d 00             lock decl 0x0(%rbp)
  496448:      75 07                   jne    496451 <arena_thread_freeres+0x771>
  49644a:      eb 1c                   jmp    496468 <arena_thread_freeres+0x788>
  49644c:      ff 4d 00                decl   0x0(%rbp)
  49644f:      74 17                   je     496468 <arena_thread_freeres+0x788>
```

```
 496451:       48 8d 7d 00             lea    0x0(%rbp),%rdi
 496455:       48 81 ec 80 00 00 00    sub    $0x80,%rsp
 49645c:       e8 cf a3 fb ff          callq  450830 <__lll_unlock_wake_private>
 496461:       48 81 c4 80 00 00 00    add    $0x80,%rsp
 496468:       85 d2                   test   %edx,%edx
 49646a:       0f 84 d8 fb ff ff       je     496048 <arena_thread_freeres+0x368>
 496470:       48 8d 3d b1 1b 00 00    lea    0x1bb1(%rip),%rdi        # 498028
<__PRETTY_FUNCTION__.12394+0x41d>
 496477:       e8 f4 38 f8 ff          callq  419d70 <malloc_printerr>
 49647c:       0f 1f 40 00             nopl   0x0(%rax)
 496480:       48 8d 3d 49 1b 00 00    lea    0x1b49(%rip),%rdi        # 497fd0
<__PRETTY_FUNCTION__.12394+0x3c5>
 496487:       e8 e4 38 f8 ff          callq  419d70 <malloc_printerr>
 49648c:       0f 1f 40 00             nopl   0x0(%rax)
 496490:       48 8b 15 d1 8c 22 00    mov    0x228cd1(%rip),%rdx        # 6bf168
<__progname>
 496497:       48 8d 05 af 0c 00 00    lea    0xcaf(%rip),%rax        # 49714d
<__PRETTY_FUNCTION__.10913+0x5d>
 49649e:       48 8d 0d d0 4a 01 00    lea    0x14ad0(%rip),%rcx        # 4aaf75
<conversion_rate+0x11d>
 4964a5:       48 8d 3d 3c 15 00 00    lea    0x153c(%rip),%rdi        # 4979e8
<__PRETTY_FUNCTION__.10520+0x128>
 4964ac:       41 b9 71 0b 00 00       mov    $0xb71,%r9d
 4964b2:       80 3a 00                cmpb   $0x0,(%rdx)
 4964b5:       48 0f 45 c8             cmovne %rax,%rcx
 4964b9:       48 83 ec 08             sub    $0x8,%rsp
 4964bd:       57                      push   %rdi
 4964be:       50                      push   %rax
 4964bf:       48 8d 05 1a 24 00 00    lea    0x241a(%rip),%rax        # 4988e0
<__PRETTY_FUNCTION__.12044>
 4964c6:       50                      push   %rax
 4964c7:       e9 f3 fd ff ff          jmpq   4962bf <arena_thread_freeres+0x5df>
 4964cc:       49 39 d0                cmp    %rdx,%r8
 4964cf:       74 56                   je     496527 <arena_thread_freeres+0x847>
 4964d1:       48 8d 74 cd 10          lea    0x10(%rbp,%rcx,8),%rsi
 4964d6:       49 89 17                mov    %rdx,(%r15)
 4964d9:      48 89 d0               mov    %rdx,%rax
 4964dc:      64 83 3c 25 18 00 00    cmpl   $0x0,%fs:0x18
 4964e3:      00 00
 4964e5:      74 01                  je     4964e8 <arena_thread_freeres+0x808>
 4964e7:      f0 4c 0f b1 06          lock cmpxchg %r8,(%rsi)
 4964ec:      48 39 d0               cmp    %rdx,%rax
 4964ef:     48 89 c1               mov    %rax,%rcx
 4964f2:     75 2e                  jne    496522 <arena_thread_freeres+0x842>
 4964f4:     e9 f2 f8 ff ff         jmpq   495deb <arena_thread_freeres+0x10b>
 4964f9:     0f 1f 80 00 00 00 00    nopl   0x0(%rax)
 496500:      49 89 0f               mov    %rcx,(%r15)
 496503:      48 89 c8               mov    %rcx,%rax
 496506:      64 83 3c 25 18 00 00    cmpl   $0x0,%fs:0x18
 49650d:      00 00
 49650f:     74 01                  je     496512 <arena_thread_freeres+0x832>
 496511:      f0 4c 0f b1 06          lock cmpxchg %r8,(%rsi)
 496516:      48 39 c1               cmp    %rax,%rcx
 496519:      0f 84 cc f8 ff ff       je     495deb <arena_thread_freeres+0x10b>
 49651f:     48 89 c1               mov    %rax,%rcx
 496522:      49 39 c8               cmp    %rcx,%r8
 496525:      75 d9                  jne    496500 <arena_thread_freeres+0x820>
 496527:      48 8d 3d 22 1b 00 00    lea    0x1b22(%rip),%rdi        # 498050
<__PRETTY_FUNCTION__.12394+0x445>
 49652e:      e8 3d 38 f8 ff          callq  419d70 <malloc_printerr>
 496533:      4d 01 d1               add    %r10,%r9
 496536:      4c 89 c8               mov    %r9,%rax
 496539:      48 83 c8 01             or     $0x1,%rax
 49653d:      49 89 40 08             mov    %rax,0x8(%r8)
 496541:      4c 89 45 60             mov    %r8,0x60(%rbp)
 496545:      e9 d2 fc ff ff          jmpq   49621c <arena_thread_freeres+0x53c>
 49654a:      8b 45 08               mov    0x8(%rbp),%eax
 49654d:      85 c0                  test   %eax,%eax
 49654f:     0f 85 1b 03 00 00       jne    496870 <arena_thread_freeres+0xb90>
 496555:      4c 39 e5               cmp    %r12,%rbp
```

```
  496558:       0f 84 2b 03 00 00       je     496889 <arena_thread_freeres+0xba9>
  49655e:       4c 8b 55 60             mov    0x60(%rbp),%r10
  496562:       4d 89 d3                mov    %r10,%r11
  496565:       49 81 e3 00 00 00 fc    and    $0xfffffffffc000000,%r11
  49656c:       49 3b 2b                cmp    (%r11),%rbp
  49656f:       0f 84 3d 01 00 00       je     4966b2 <arena_thread_freeres+0x9d2>
  496575:       48 8b 15 ec 8b 22 00    mov    0x228bec(%rip),%rdx        # 6bf168
<__progname>
  49657c:       48 8d 05 ca 0b 00 00    lea    0xbca(%rip),%rax        # 49714d
<__PRETTY_FUNCTION__.10913+0x5d>
  496583:       48 8d 0d eb 49 01 00    lea    0x149eb(%rip),%rcx        # 4aaf75
<conversion_rate+0x11d>
  49658a:       48 8d 3d 8d 14 00 00    lea    0x148d(%rip),%rdi        # 497a1e
<__PRETTY_FUNCTION__.10520+0x15e>
  496591:       41 b9 17 11 00 00       mov    $0x1117,%r9d
  496597:       80 3a 00                cmpb   $0x0,(%rdx)
  49659a:       48 0f 45 c8             cmovne %rax,%rcx
  49659e:       48 83 ec 08             sub    $0x8,%rsp
  4965a2:       57                      push   %rdi
  4965a3:       50                      push   %rax
  4965a4:       48 8d 05 05 23 00 00    lea    0x2305(%rip),%rax        # 4988b0
<__PRETTY_FUNCTION__.12320>
  4965ab:       50                      push   %rax
  4965ac:       e9 0e fd ff ff          jmpq   4962bf <arena_thread_freeres+0x5df>
  4965b1:       48 8d 3d 28 1b 00 00    lea    0x1b28(%rip),%rdi        # 4980e0
<__PRETTY_FUNCTION__.12394+0x4d5>
  4965b8:       e8 b3 37 f8 ff          callq  419d70 <malloc_printerr>
  4965bd:       48 b9 01 01 01 01 01    movabs $0x101010101010101,%rcx
  4965c4:       01 01 01
  4965c7:       0f b6 c0                movzbl %al,%eax
  4965ca:       49 8d 51 f0             lea    -0x10(%r9),%rdx
  4965ce:       48 0f af c1             imul   %rcx,%rax
  4965d2:       49 8d 7f 08             lea    0x8(%r15),%rdi
  4965d6:       89 d1                   mov    %edx,%ecx
  4965d8:       48 83 e7 f8             and    $0xfffffffffffffff8,%rdi
  4965dc:       49 89 07                mov    %rax,(%r15)
  4965df:       49 89 44 0f f8          mov    %rax,-0x8(%r15,%rcx,1)
  4965e4:       4c 89 f9                mov    %r15,%rcx
  4965e7:       48 29 f9                sub    %rdi,%rcx
  4965ea:       01 ca                   add    %ecx,%edx
  4965ec:       c1 ea 03                shr    $0x3,%edx
  4965ef:       89 d1                   mov    %edx,%ecx
  4965f1:       f3 48 ab                rep stos %rax,%es:(%rdi)
  4965f4:       e9 5d fa ff ff          jmpq   496056 <arena_thread_freeres+0x376>
  4965f9:       44 89 ee                mov    %r13d,%esi
  4965fc:       31 c0                   xor    %eax,%eax
  4965fe:       83 3d 47 f1 22 00 00    cmpl   $0x0,0x22f147(%rip)        # 6c574c
<__libc_multiple_threads>
  496605:       74 09                   je     496610 <arena_thread_freeres+0x930>
  496607:       f0 0f b1 75 00          lock cmpxchg %esi,0x0(%rbp)
  49660c:       75 08                   jne    496616 <arena_thread_freeres+0x936>
  49660e:       eb 1d                   jmp    49662d <arena_thread_freeres+0x94d>
  496610:       0f b1 75 00             cmpxchg %esi,0x0(%rbp)
  496614:       74 17                   je     49662d <arena_thread_freeres+0x94d>
  496616:       48 8d 7d 00             lea    0x0(%rbp),%rdi
  49661a:       48 81 ec 80 00 00 00    sub    $0x80,%rsp
  496621:       e8 da a1 fb ff          callq  450800 <__lll_lock_wait_private>
  496626:       48 81 c4 80 00 00 00    add    $0x80,%rsp
  49662d:       45 31 f6                xor    %r14d,%r14d
  496630:       e9 97 fa ff ff          jmpq   4960cc <arena_thread_freeres+0x3ec>
  496635:       48 8d 3d 3c 1a 00 00    lea    0x1a3c(%rip),%rdi        # 498078
<__PRETTY_FUNCTION__.12394+0x46d>
  49663c:       e8 2f 37 f8 ff          callq  419d70 <malloc_printerr>
  496641:       48 8b 50 08             mov    0x8(%rax),%rdx
  496645:       48 83 e2 f8             and    $0xfffffffffffffff8,%rdx
  496649:       48 01 d0                add    %rdx,%rax
  49664c:       48 39 c1                cmp    %rax,%rcx
  49664f:       0f 82 92 fa ff ff       jb     4960e7 <arena_thread_freeres+0x407>
  496655:       48 8d 3d 3c 1a 00 00    lea    0x1a3c(%rip),%rdi        # 498098
<__PRETTY_FUNCTION__.12394+0x48d>
  49665c:       e8 0f 37 f8 ff          callq  419d70 <malloc_printerr>
  496661:	48 8d 3d 85 12 00 00 	lea    0x1285(%rip),%rdi        # 4978ed
<__PRETTY_FUNCTION__.10520+0x2d>
  496668:	e8 03 37 f8 ff       	callq  419d70 <malloc_printerr>
  49666d:	48 8d 3d 44 1a 00 00 	lea    0x1a44(%rip),%rdi        # 4980b8
<__PRETTY_FUNCTION__.12394+0x4ad>
  496674:	e8 f7 36 f8 ff       	callq  419d70 <malloc_printerr>
  496679:	49 8d 51 f0          	lea    -0x10(%r9),%rdx
  49667d:	4c 89 ff             	mov    %r15,%rdi
  496680:	4c 89 54 24 38       	mov    %r10,0x38(%rsp)
  496685:	48 89 4c 24 30       	mov    %rcx,0x30(%rsp)
  49668a:	4c 89 44 24 28       	mov    %r8,0x28(%rsp)
  49668f:	4c 89 4c 24 20       	mov    %r9,0x20(%rsp)
  496694:	e8 0f 9e f6 ff       	callq  4004a8 <.plt+0x78>
  496699:	4c 8b 54 24 38       	mov    0x38(%rsp),%r10
  49669e:	48 8b 4c 24 30       	mov    0x30(%rsp),%rcx
  4966a3:	4c 8b 44 24 28       	mov    0x28(%rsp),%r8
  4966a8:	4c 8b 4c 24 20       	mov    0x20(%rsp),%r9
  4966ad:	e9 6d fa ff ff       	jmpq   49611f <arena_thread_freeres+0x43f>
  4966b2:	4d 8d 4b 20          	lea    0x20(%r11),%r9
  4966b6:	48 8b 0d 2b 81 22 00 	mov    0x22812b(%rip),%rcx        # 6be7e8
<mp_+0x8>
  4966bd:	48 8b 15 34 8b 22 00 	mov    0x228b34(%rip),%rdx        # 6bf1f8
<_dl_pagesize>
  4966c4:	4d 39 d1             	cmp    %r10,%r9
  4966c7:	0f 85 fe 03 00 00    	jne    496acb <arena_thread_freeres+0xdeb>
  4966cd:	4d 8b 43 08          	mov    0x8(%r11),%r8
  4966d1:	49 8b 78 10          	mov    0x10(%r8),%rdi
  4966d5:	48 8d 47 f0          	lea    -0x10(%rdi),%rax
  4966d9:	49 8d 34 00          	lea    (%r8,%rax,1),%rsi
  4966dd:	83 e6 0f             	and    $0xf,%esi
  4966e0:	48 29 f0             	sub    %rsi,%rax
  4966e3:	4c 01 c0             	add    %r8,%rax
  4966e6:	48 83 78 08 01       	cmpq   $0x1,0x8(%rax)
  4966eb:	0f 85 04 02 00 00    	jne    4968f5 <arena_thread_freeres+0xc15>
  4966f1:	4c 8d 54 11 20       	lea    0x20(%rcx,%rdx,1),%r10
```

```
 4966f6:       48 89 5c 24 20          mov    %rbx,0x20(%rsp)
 4966fb:       4c 89 54 24 28          mov    %r10,0x28(%rsp)
 496700:       4c 8d 52 ff             lea    -0x1(%rdx),%r10
 496704:       4c 89 54 24 30          mov    %r10,0x30(%rsp)
 496709:       48 2b 00                sub    (%rax),%rax
 49670c:       49 89 c7                mov    %rax,%r15
 49670f:       48 8b 40 08             mov    0x8(%rax),%rax
 496713:       48 89 c3                mov    %rax,%rbx
 496716:       48 83 e3 f8             and    $0xfffffffffffffff8,%rbx
 49671a:       48 01 de                add    %rbx,%rsi
 49671d:       48 8d 5e 0f             lea    0xf(%rsi),%rbx
 496721:       48 83 fb 3e             cmp    $0x3e,%rbx
 496725:       0f 87 1f 02 00 00       ja     49694a <arena_thread_freeres+0xc6a>
 49672b:       a8 01                   test   $0x1,%al
 49672d:       48 8d 5e 10             lea    0x10(%rsi),%rbx
 496731:       75 03                   jne    496736 <arena_thread_freeres+0xa56>
 496733:       49 03 1f                add    (%r15),%rbx
 496736:       48 8d 43 ff             lea    -0x1(%rbx),%rax
 49673a:       48 3d fe ff ff 03       cmp    $0x3fffffe,%rax
 496740:       0f 87 7b 02 00 00       ja     4969c1 <arena_thread_freeres+0xce1>
 496746:       b8 00 00 00 04          mov    $0x4000000,%eax
 49674b:       48 29 f8                sub    %rdi,%rax
 49674e:       48 01 d8                add    %rbx,%rax
 496751:       48 3b 44 24 28          cmp    0x28(%rsp),%rax
 496756:       0f 82 67 03 00 00       jb     496ac3 <arena_thread_freeres+0xde3>
 49675c:       49 8b 43 10             mov    0x10(%r11),%rax
 496760:       48 29 85 88 08 00 00    sub    %rax,0x888(%rbp)
 496767:       90                      nop
 496768:       49 8d 83 00 00 00 04    lea    0x4000000(%r11),%rax
 49676f:       48 39 05 a2 a6 22 00    cmp    %rax,0x22a6a2(%rip)        # 6c0e18
<aligned_heap_area>
 496776:       75 0b                   jne    496783 <arena_thread_freeres+0xaa3>
 496778:       48 c7 05 95 a6 22 00    movq   $0x0,0x22a695(%rip)        # 6c0e18
<aligned_heap_area>
```

```
49677f:      00 00 00 00
496783:       be 00 00 00 04          mov    $0x4000000,%esi
496788:       4c 89 df               mov    %r11,%rdi
49678b:       48 89 54 24 48          mov    %rdx,0x48(%rsp)
496790:       48 89 4c 24 40          mov    %rcx,0x40(%rsp)
496795:       4c 89 44 24 38          mov    %r8,0x38(%rsp)
49679a:       e8 c1 88 fb ff          callq  44f060 <__munmap>
49679f:      41 f6 47 08 01          testb  $0x1,0x8(%r15)
4967a4:       4c 8b 44 24 38          mov    0x38(%rsp),%r8
4967a9:       48 8b 4c 24 40          mov    0x40(%rsp),%rcx
4967ae:       48 8b 54 24 48          mov    0x48(%rsp),%rdx
4967b3:       75 52                  jne    496807 <arena_thread_freeres+0xb27>
4967b5:       4d 2b 3f               sub    (%r15),%r15
4967b8:       49 8b 7f 08             mov    0x8(%r15),%rdi
4967bc:       48 89 f8               mov    %rdi,%rax
4967bf:      48 83 e0 f8             and    $0xfffffffffffffff8,%rax
4967c3:       49 3b 04 07             cmp    (%r15,%rax,1),%rax
4967c7:       0f 85 b0 00 00 00       jne    49687d <arena_thread_freeres+0xb9d>
4967cd:       49 8b 47 10             mov    0x10(%r15),%rax
4967d1:       4c 3b 78 18             cmp    0x18(%rax),%r15
4967d5:       0f 85 86 fe ff ff       jne    496661 <arena_thread_freeres+0x981>
4967db:       49 8b 77 18             mov    0x18(%r15),%rsi
4967df:      4c 3b 7e 10             cmp    0x10(%rsi),%r15
4967e3:       0f 85 78 fe ff ff       jne    496661 <arena_thread_freeres+0x981>
4967e9:       48 81 ff ff 03 00 00    cmp    $0x3ff,%rdi
4967f0:      48 89 70 18             mov    %rsi,0x18(%rax)
4967f4:      48 89 46 10             mov    %rax,0x10(%rsi)
4967f8:      76 0d                  jbe    496807 <arena_thread_freeres+0xb27>
4967fa:      49 8b 77 20             mov    0x20(%r15),%rsi
4967fe:      48 85 f6               test   %rsi,%rsi
496801:       0f 85 fc 04 00 00       jne    496d03 <arena_thread_freeres+0x1023>
496807:       49 8d 04 1f             lea    (%r15,%rbx,1),%rax
49680b:       48 85 44 24 30          test   %rax,0x30(%rsp)
496810:       0f 85 6c 02 00 00       jne    496a82 <arena_thread_freeres+0xda2>
```

```
 496816:	49 8b 70 10          	mov    0x10(%r8),%rsi
 49681a:	4c 01 c6             	add    %r8,%rsi
 49681d:	48 39 f0             	cmp    %rsi,%rax
 496820:	0f 84 11 02 00 00    	je     496a37 <arena_thread_freeres+0xd57>
 496826:	48 8b 15 3b 89 22 00 	mov    0x22893b(%rip),%rdx        # 6bf168
<__progname>
 49682d:	80 3a 00             	cmpb   $0x0,(%rdx)
 496830:	0f 84 4c 03 00 00    	je     496b82 <arena_thread_freeres+0xea2>
 496836:	48 8d 05 10 09 00 00 	lea    0x910(%rip),%rax        # 49714d
<__PRETTY_FUNCTION__.10913+0x5d>
 49683d:	48 89 c1             	mov    %rax,%rcx
 496840:	48 8d 3d b9 19 00 00 	lea    0x19b9(%rip),%rdi        # 498200
<__PRETTY_FUNCTION__.12394+0x5f5>
 496847:	48 83 ec 08          	sub    $0x8,%rsp
 49684b:	41 b9 77 02 00 00    	mov    $0x277,%r9d
 496851:	4c 8d 05 dc 10 00 00 	lea    0x10dc(%rip),%r8        # 497934
<__PRETTY_FUNCTION__.10520+0x74>
 496858:	57                   	push   %rdi
 496859:	50                   	push   %rax
 49685a:	48 8d 05 3f 20 00 00 	lea    0x203f(%rip),%rax        # 4988a0
<__PRETTY_FUNCTION__.11706>
 496861:	50                   	push   %rax
 496862:	e9 5f fa ff ff       	jmpq   4962c6 <arena_thread_freeres+0x5e6>
 496867:	66 0f 1f 84 00 00 00 	nopw   0x0(%rax,%rax,1)
 49686e:	00 00 
 496870:	48 89 ef             	mov    %rbp,%rdi
 496873:	e8 18 35 f8 ff       	callq  419d90 <malloc_consolidate>
 496878:	e9 d8 fc ff ff       	jmpq   496555 <arena_thread_freeres+0x875>
 49687d:	48 8d 3d 4c 10 00 00 	lea    0x104c(%rip),%rdi        # 4978d0
<__PRETTY_FUNCTION__.10520+0x10>
 496884:	e8 e7 34 f8 ff       	callq  419d70 <malloc_printerr>
 496889:	48 8b 45 60          	mov    0x60(%rbp),%rax
 49688d:	48 8b 40 08          	mov    0x8(%rax),%rax
 496891:	48 83 e0 f8          	and    $0xfffffffffffffff8,%rax
 496895:	48 3b 05 44 7f 22 00 	cmp    0x227f44(%rip),%rax        # 6be7e0 <mp_>
 49689c:	0f 82 87 f9 ff ff    	jb     496229 <arena_thread_freeres+0x549>
 4968a2:       48 8b 3d 3f 7f 22 00    mov    0x227f3f(%rip),%rdi        # 6be7e8
<mp_+0x8>
 4968a9:       e8 e2 3e f8 ff          callq  41a790 <systrim.isra.1.constprop.11>
 4968ae:       e9 76 f9 ff ff          jmpq   496229 <arena_thread_freeres+0x549>
 4968b3:       64 48 83 3c 25 c8 ff    cmpq   $0x0,%fs:0xffffffffffffffc8
 4968ba:       ff ff 00
 4968bd:       0f 84 3b 01 00 00       je     4969fe <arena_thread_freeres+0xd1e>
 4968c3:       a8 04                   test   $0x4,%al
 4968c5:       48 8d 3d b4 7f 22 00    lea    0x227fb4(%rip),%rdi        # 6be880
<main_arena>
 4968cc:       74 0c                   je     4968da <arena_thread_freeres+0xbfa>
 4968ce:       48 89 d8                mov    %rbx,%rax
 4968d1:       48 25 00 00 00 fc       and    $0xfffffffffc000000,%rax
 4968d7:       48 8b 38                mov    (%rax),%rdi
 4968da:       31 d2                   xor    %edx,%edx
 4968dc:       48 89 de                mov    %rbx,%rsi
 4968df:       e8 dc 56 f8 ff          callq  41bfc0 <_int_free>
 4968e4:       e9 7f f5 ff ff          jmpq   495e68 <arena_thread_freeres+0x188>
 4968e9:       48 8d 3d 18 18 00 00    lea    0x1818(%rip),%rdi        # 498108
<__PRETTY_FUNCTION__.12394+0x4fd>
 4968f0:       e8 7b 34 f8 ff          callq  419d70 <malloc_printerr>
 4968f5:       48 8b 15 6c 88 22 00    mov    0x22886c(%rip),%rdx        # 6bf168
<__progname>
 4968fc:       48 8d 05 4a 08 00 00    lea    0x84a(%rip),%rax        # 49714d
<__PRETTY_FUNCTION__.10913+0x5d>
 496903:       48 8d 0d 6b 46 01 00    lea    0x1466b(%rip),%rcx        # 4aaf75
<conversion_rate+0x11d>
 49690a:       48 8d 3d 1f 18 00 00    lea    0x181f(%rip),%rdi        # 498130
<__PRETTY_FUNCTION__.12394+0x525>
 496911:       41 b9 64 02 00 00       mov    $0x264,%r9d
 496917:       4c 8d 05 16 10 00 00    lea    0x1016(%rip),%r8        # 497934
<__PRETTY_FUNCTION__.10520+0x74>
 49691e:       80 3a 00                cmpb   $0x0,(%rdx)
 496921:       48 0f 45 c8             cmovne %rax,%rcx
 496925:       48 83 ec 08             sub    $0x8,%rsp
 496929:       57                      push   %rdi
 49692a:       50                      push   %rax
 49692b:	48 8d 05 6e 1f 00 00 	lea    0x1f6e(%rip),%rax        # 4988a0
<__PRETTY_FUNCTION__.11706>

 496932:	50                   	push   %rax

 496933:	e9 8e f9 ff ff       	jmpq   4962c6 <arena_thread_freeres+0x5e6>

 496938:	48 8b b4 24 98 00 00 	mov    0x98(%rsp),%rsi

 49693f:	00

 496940:	4c 89 ef             	mov    %r13,%rdi

 496943:	ff d0                	callq  *%rax

 496945:	e9 1e f5 ff ff       	jmpq   495e68 <arena_thread_freeres+0x188>

 49694a:	48 8b 15 17 88 22 00 	mov    0x228817(%rip),%rdx        # 6bf168
<__progname>

 496951:	48 8d 05 f5 07 00 00 	lea    0x7f5(%rip),%rax        # 49714d
<__PRETTY_FUNCTION__.10913+0x5d>

 496958:	48 8d 0d 16 46 01 00 	lea    0x14616(%rip),%rcx        # 4aaf75
<conversion_rate+0x11d>

 49695f:	48 8d 3d fa 17 00 00 	lea    0x17fa(%rip),%rdi        # 498160
<__PRETTY_FUNCTION__.12394+0x555>

 496966:	41 b9 67 02 00 00    	mov    $0x267,%r9d

 49696c:	4c 8d 05 c1 0f 00 00 	lea    0xfc1(%rip),%r8        # 497934
<__PRETTY_FUNCTION__.10520+0x74>

 496973:	80 3a 00             	cmpb   $0x0,(%rdx)

 496976:	48 0f 45 c8          	cmovne %rax,%rcx

 49697a:	48 83 ec 08          	sub    $0x8,%rsp

 49697e:	57                   	push   %rdi

 49697f:	50                   	push   %rax

 496980:	48 8d 05 19 1f 00 00 	lea    0x1f19(%rip),%rax        # 4988a0
<__PRETTY_FUNCTION__.11706>

 496987:	50                   	push   %rax

 496988:	e9 39 f9 ff ff       	jmpq   4962c6 <arena_thread_freeres+0x5e6>

 49698d:	48 3b 48 28          	cmp    0x28(%rax),%rcx

 496991:	0f 85 94 00 00 00    	jne    496a2b <arena_thread_freeres+0xd4b>

 496997:	48 8b 71 28          	mov    0x28(%rcx),%rsi

 49699b:	48 3b 4e 20          	cmp    0x20(%rsi),%rcx

 49699f:	0f 85 86 00 00 00    	jne    496a2b <arena_thread_freeres+0xd4b>

 4969a5:	48 83 7a 20 00       	cmpq   $0x0,0x20(%rdx)

 4969aa:	0f 84 07 02 00 00    	je     496bb7 <arena_thread_freeres+0xed7>
 4969b0:	48 89 70 28          	mov    %rsi,0x28(%rax)
 4969b4:	48 8b 51 28          	mov    0x28(%rcx),%rdx
 4969b8:	48 89 42 20          	mov    %rax,0x20(%rdx)
 4969bc:	e9 19 fa ff ff       	jmpq   4963da <arena_thread_freeres+0x6fa>
 4969c1:	48 8b 15 a0 87 22 00 	mov    0x2287a0(%rip),%rdx        # 6bf168
<__progname>
 4969c8:	80 3a 00             	cmpb   $0x0,(%rdx)
 4969cb:	74 4e                	je     496a1b <arena_thread_freeres+0xd3b>
 4969cd:	48 8d 05 79 07 00 00 	lea    0x779(%rip),%rax        # 49714d
<__PRETTY_FUNCTION__.10913+0x5d>
 4969d4:	48 89 c1             	mov    %rax,%rcx
 4969d7:	48 8d 3d b2 17 00 00 	lea    0x17b2(%rip),%rdi        # 498190
<__PRETTY_FUNCTION__.12394+0x585>
 4969de:	48 83 ec 08          	sub    $0x8,%rsp
 4969e2:	41 b9 6a 02 00 00    	mov    $0x26a,%r9d
 4969e8:	4c 8d 05 45 0f 00 00 	lea    0xf45(%rip),%r8        # 497934
<__PRETTY_FUNCTION__.10520+0x74>
 4969ef:	57                   	push   %rdi
 4969f0:	50                   	push   %rax
 4969f1:	48 8d 05 a8 1e 00 00 	lea    0x1ea8(%rip),%rax        # 4988a0
<__PRETTY_FUNCTION__.11706>
 4969f8:	50                   	push   %rax
 4969f9:	e9 c8 f8 ff ff       	jmpq   4962c6 <arena_thread_freeres+0x5e6>
 4969fe:	64 80 3c 25 d0 ff ff 	cmpb   $0x0,%fs:0xffffffffffffffd0
 496a05:	ff 00
 496a07:	0f 85 b6 fe ff ff    	jne    4968c3 <arena_thread_freeres+0xbe3>
 496a0d:	e8 4e 8d f8 ff       	callq  41f760 <tcache_init.part.4>
 496a12:	49 8b 45 f8          	mov    -0x8(%r13),%rax
 496a16:	e9 a8 fe ff ff       	jmpq   4968c3 <arena_thread_freeres+0xbe3>
 496a1b:	48 8d 0d 53 45 01 00 	lea    0x14553(%rip),%rcx        # 4aaf75
<conversion_rate+0x11d>
 496a22:	48 8d 05 24 07 00 00 	lea    0x724(%rip),%rax        # 49714d
<__PRETTY_FUNCTION__.10913+0x5d>
 496a29:	eb ac                	jmp    4969d7 <arena_thread_freeres+0xcf7>
 496a2b:	48 8d 3d 16 12 00 00 	lea    0x1216(%rip),%rdi        # 497c48
<__PRETTY_FUNCTION__.12394+0x3d>
 496a32:	e8 39 33 f8 ff       	callq  419d70 <malloc_printerr>
 496a37:    49 8d 40 20           lea    0x20(%r8),%rax
 496a3b:    48 83 cb 01           or     $0x1,%rbx
 496a3f:    4c 89 7d 60           mov    %r15,0x60(%rbp)
 496a43:    49 89 5f 08           mov    %rbx,0x8(%r15)
 496a47:    49 39 c7              cmp    %rax,%r15
 496a4a:    0f 85 a5 01 00 00     jne    496bf5 <arena_thread_freeres+0xf15>
 496a50:    49 8b 58 08           mov    0x8(%r8),%rbx
 496a54:    4d 89 c3              mov    %r8,%r11
 496a57:    4d 89 f9              mov    %r15,%r9
 496a5a:    48 8b 7b 10           mov    0x10(%rbx),%rdi
 496a5e:    48 8d 47 f0           lea    -0x10(%rdi),%rax
 496a62:    48 8d 34 03           lea    (%rbx,%rax,1),%rsi
 496a66:    83 e6 0f              and    $0xf,%esi
 496a69:    48 29 f0              sub    %rsi,%rax
 496a6c:    48 01 d8              add    %rbx,%rax
 496a6f:    48 83 78 08 01        cmpq   $0x1,0x8(%rax)
 496a74:    0f 85 7b fe ff ff     jne    4968f5 <arena_thread_freeres+0xc15>
 496a7a:    49 89 d8              mov    %rbx,%r8
 496a7d:    e9 87 fc ff ff        jmpq   496709 <arena_thread_freeres+0xa29>
 496a82:    48 8b 15 df 86 22 00  mov    0x2286df(%rip),%rdx        # 6bf168
<__progname>
 496a89:    80 3a 00              cmpb   $0x0,(%rdx)
 496a8c:    0f 84 50 01 00 00     je     496be2 <arena_thread_freeres+0xf02>
 496a92:    48 8d 05 b4 06 00 00  lea    0x6b4(%rip),%rax        # 49714d
<__PRETTY_FUNCTION__.10913+0x5d>
 496a99:    48 89 c1              mov    %rax,%rcx
 496a9c:    48 8d 3d 1d 17 00 00  lea    0x171d(%rip),%rdi        # 4981c0
<__PRETTY_FUNCTION__.12394+0x5b5>
 496aa3:    48 83 ec 08           sub    $0x8,%rsp
 496aa7:    41 b9 76 02 00 00     mov    $0x276,%r9d
 496aad:    4c 8d 05 80 0e 00 00  lea    0xe80(%rip),%r8        # 497934
<__PRETTY_FUNCTION__.10520+0x74>
 496ab4:    57                    push   %rdi
 496ab5:    50                    push   %rax
 496ab6:    48 8d 05 e3 1d 00 00  lea    0x1de3(%rip),%rax        # 4988a0
<__PRETTY_FUNCTION__.11706>
```

```
  496abd:       50                      push   %rax
  496abe:       e9 03 f8 ff ff          jmpq   4962c6 <arena_thread_freeres+0x5e6>
  496ac3:       48 8b 5c 24 20          mov    0x20(%rsp),%rbx
  496ac8:       4d 89 ca                mov    %r9,%r10
  496acb:       49 8b 42 08             mov    0x8(%r10),%rax
  496acf:       48 83 e0 f8             and    $0xfffffffffffffff8,%rax
  496ad3:       48 39 05 06 7d 22 00    cmp    %rax,0x227d06(%rip)        # 6be7e0 <mp_>
  496ada:       48 89 44 24 28          mov    %rax,0x28(%rsp)
  496adf:       0f 87 44 f7 ff ff       ja     496229 <arena_thread_freeres+0x549>
  496ae5:       48 83 e8 21             sub    $0x21,%rax
  496ae9:       0f 88 3a f7 ff ff       js     496229 <arena_thread_freeres+0x549>
  496aef:       48 39 c1                cmp    %rax,%rcx
  496af2:       0f 83 31 f7 ff ff       jae    496229 <arena_thread_freeres+0x549>
  496af8:       48 29 c8                sub    %rcx,%rax
  496afb:       48 f7 da                neg    %rdx
  496afe:       48 21 d0                and    %rdx,%rax
  496b01:       48 89 44 24 20          mov    %rax,0x20(%rsp)
  496b06:       0f 84 1d f7 ff ff       je     496229 <arena_thread_freeres+0x549>
  496b0c:       4d 8b 7b 10             mov    0x10(%r11),%r15
  496b10:       49 29 c7                sub    %rax,%r15
  496b13:       49 83 ff 1f             cmp    $0x1f,%r15
  496b17:       0f 8e 0c f7 ff ff       jle    496229 <arena_thread_freeres+0x549>
  496b1d:       83 3d 9c 7c 22 00 00    cmpl   $0x0,0x227c9c(%rip)        # 6be7c0
<may_shrink_heap.10708>
  496b24:       0f 88 26 01 00 00       js     496c50 <arena_thread_freeres+0xf70>
  496b2a:       0f 95 c0                setne  %al
  496b2d:       84 c0                   test   %al,%al
  496b2f:       4b 8d 3c 3b             lea    (%r11,%r15,1),%rdi
  496b33:       4c 89 54 24 38          mov    %r10,0x38(%rsp)
  496b38:       4c 89 5c 24 30          mov    %r11,0x30(%rsp)
  496b3d:       0f 85 d8 00 00 00       jne    496c1b <arena_thread_freeres+0xf3b>
  496b43:       48 8b 74 24 20          mov    0x20(%rsp),%rsi
  496b48:       ba 04 00 00 00          mov    $0x4,%edx
  496b4d:       e8 6e 85 fb ff          callq  44f0c0 <__madvise>
  496b52:       4c 8b 54 24 38          mov    0x38(%rsp),%r10
  496b57:       4c 8b 5c 24 30          mov    0x30(%rsp),%r11
  496b5c:       4d 89 7b 10             mov    %r15,0x10(%r11)
  496b60:       90                      nop
  496b61:       48 8b 4c 24 20          mov    0x20(%rsp),%rcx
  496b66:       48 8b 44 24 28          mov    0x28(%rsp),%rax
  496b6b:       48 29 8d 88 08 00 00    sub    %rcx,0x888(%rbp)
  496b72:       48 29 c8                sub    %rcx,%rax
  496b75:       48 83 c8 01             or     $0x1,%rax
  496b79:       49 89 42 08             mov    %rax,0x8(%r10)
  496b7d:       e9 a7 f6 ff ff          jmpq   496229 <arena_thread_freeres+0x549>
  496b82:       48 8d 0d ec 43 01 00    lea    0x143ec(%rip),%rcx        # 4aaf75
<conversion_rate+0x11d>
  496b89:       48 8d 05 bd 05 00 00    lea    0x5bd(%rip),%rax        # 49714d
<__PRETTY_FUNCTION__.10913+0x5d>
  496b90:       e9 ab fc ff ff          jmpq   496840 <arena_thread_freeres+0xb60>
  496b95:       49 39 d0                cmp    %rdx,%r8
  496b98:       74 74                   je     496c0e <arena_thread_freeres+0xf2e>
  496b9a:       48 89 50 20             mov    %rdx,0x20(%rax)
  496b9e:       49 8b 50 20             mov    0x20(%r8),%rdx
  496ba2:       48 89 70 28             mov    %rsi,0x28(%rax)
  496ba6:       48 89 42 28             mov    %rax,0x28(%rdx)
  496baa:       49 8b 50 28             mov    0x28(%r8),%rdx
  496bae:       48 89 42 20             mov    %rax,0x20(%rdx)
  496bb2:       e9 f9 f5 ff ff          jmpq   4961b0 <arena_thread_freeres+0x4d0>
  496bb7:       48 39 c1                cmp    %rax,%rcx
  496bba:       0f 84 73 01 00 00       je     496d33 <arena_thread_freeres+0x1053>
  496bc0:       48 89 42 20             mov    %rax,0x20(%rdx)
  496bc4:       48 8b 41 20             mov    0x20(%rcx),%rax
  496bc8:       48 89 72 28             mov    %rsi,0x28(%rdx)
  496bcc:       48 89 50 28             mov    %rdx,0x28(%rax)
  496bd0:       48 8b 41 28             mov    0x28(%rcx),%rax
  496bd4:       48 89 50 20             mov    %rdx,0x20(%rax)
  496bd8:       e9 fd f7 ff ff          jmpq   4963da <arena_thread_freeres+0x6fa>
  496bdd:       e8 3e 9d fb ff          callq  450920 <__stack_chk_fail>
  496be2:       48 8d 0d 8c 43 01 00    lea    0x1438c(%rip),%rcx        # 4aaf75
<conversion_rate+0x11d>
  496be9:       48 8d 05 5d 05 00 00    lea    0x55d(%rip),%rax        # 49714d
<__PRETTY_FUNCTION__.10913+0x5d>
  496bf0:       e9 a7 fe ff ff          jmpq   496a9c <arena_thread_freeres+0xdbc>
  496bf5:       4d 89 fa                mov    %r15,%r10
  496bf8:       48 8b 5c 24 20          mov    0x20(%rsp),%rbx
  496bfd:       4d 89 c3                mov    %r8,%r11
  496c00:       e9 c6 fe ff ff          jmpq   496acb <arena_thread_freeres+0xdeb>
  496c05:       48 83 e2 f8             and    $0xfffffffffffffff8,%rdx
  496c09:       e9 9c f1 ff ff          jmpq   495daa <arena_thread_freeres+0xca>
  496c0e:       48 89 40 28             mov    %rax,0x28(%rax)
  496c12:       48 89 40 20             mov    %rax,0x20(%rax)
  496c16:       e9 95 f5 ff ff          jmpq   4961b0 <arena_thread_freeres+0x4d0>
  496c1b:       48 8b 74 24 20          mov    0x20(%rsp),%rsi
  496c20:       45 31 c9                xor    %r9d,%r9d
  496c23:       41 83 c8 ff             or     $0xffffffff,%r8d
  496c27:       31 d2                   xor    %edx,%edx
  496c29:       b9 32 00 00 00          mov    $0x32,%ecx
  496c2e:       e8 4d 83 fb ff          callq  44ef80 <__mmap64>
  496c33:       48 83 c0 01             add    $0x1,%rax
  496c37:       0f 84 ec f5 ff ff       je     496229 <arena_thread_freeres+0x549>
  496c3d:       4c 8b 5c 24 30          mov    0x30(%rsp),%r11
  496c42:       4c 8b 54 24 38          mov    0x38(%rsp),%r10
  496c47:       4d 89 7b 18             mov    %r15,0x18(%r11)
  496c4b:       e9 0c ff ff ff          jmpq   496b5c <arena_thread_freeres+0xe7c>
  496c50:       8b 0d 62 6e 22 00       mov    0x226e62(%rip),%ecx        # 6bdab8
<__libc_enable_secure>
  496c56:       85 c9                   test   %ecx,%ecx
  496c58:       89 0d 62 7b 22 00       mov    %ecx,0x227b62(%rip)        # 6be7c0
<may_shrink_heap.10708>
  496c5e:       89 4c 24 48             mov    %ecx,0x48(%rsp)
  496c62:       75 2b                   jne    496c8f <arena_thread_freeres+0xfaf>
  496c64:       48 8d 3d cd 15 00 00    lea    0x15cd(%rip),%rdi        # 498238
<__PRETTY_FUNCTION__.12394+0x62d>
```

```
  496c6b:       31 c0                   xor    %eax,%eax
  496c6d:       be 00 00 08 00          mov    $0x80000,%esi
  496c72:       4c 89 54 24 38          mov    %r10,0x38(%rsp)
  496c77:       4c 89 5c 24 30          mov    %r11,0x30(%rsp)
  496c7c:       e8 5f 75 fb ff          callq  44e1e0 <__open64_nocancel>
  496c81:       85 c0                   test   %eax,%eax
  496c83:       4c 8b 5c 24 30          mov    0x30(%rsp),%r11
  496c88:       4c 8b 54 24 38          mov    0x38(%rsp),%r10
  496c8d:       79 0f                   jns    496c9e <arena_thread_freeres+0xfbe>
  496c8f:       83 3d 2a 7b 22 00 00    cmpl   $0x0,0x227b2a(%rip)        # 6be7c0
<may_shrink_heap.10708>
  496c96:       0f 95 c0                setne  %al
  496c99:       e9 8f fe ff ff          jmpq   496b2d <arena_thread_freeres+0xe4d>
  496c9e:       48 8b 74 24 18          mov    0x18(%rsp),%rsi
  496ca3:       ba 01 00 00 00          mov    $0x1,%edx
  496ca8:       89 c7                   mov    %eax,%edi
  496caa:       4c 89 54 24 40          mov    %r10,0x40(%rsp)
  496caf:       4c 89 5c 24 38          mov    %r11,0x38(%rsp)
  496cb4:       89 44 24 30             mov    %eax,0x30(%rsp)
  496cb8:       e8 63 76 fb ff          callq  44e320 <__read_nocancel>
  496cbd:       48 85 c0                test   %rax,%rax
  496cc0:       44 8b 44 24 30          mov    0x30(%rsp),%r8d
  496cc5:       4c 8b 5c 24 38          mov    0x38(%rsp),%r11
  496cca:       4c 8b 54 24 40          mov    0x40(%rsp),%r10
  496ccf:       8b 4c 24 48             mov    0x48(%rsp),%ecx
  496cd3:       7e 0a                   jle    496cdf <arena_thread_freeres+0xfff>
  496cd5:       31 c9                   xor    %ecx,%ecx
  496cd7:       80 7c 24 57 32          cmpb   $0x32,0x57(%rsp)
  496cdc:       0f 94 c1                sete   %cl
  496cdf:       44 89 c7                mov    %r8d,%edi
  496ce2:       4c 89 54 24 38          mov    %r10,0x38(%rsp)
  496ce7:       4c 89 5c 24 30          mov    %r11,0x30(%rsp)
  496cec:       89 0d ce 7a 22 00       mov    %ecx,0x227ace(%rip)        # 6be7c0
<may_shrink_heap.10708>
```

```
 496cf2:	e8 a9 79 fb ff       	callq  44e6a0 <__close_nocancel>
 496cf7:	4c 8b 54 24 38       	mov    0x38(%rsp),%r10
 496cfc:	4c 8b 5c 24 30       	mov    0x30(%rsp),%r11
 496d01:	eb 8c                	jmp    496c8f <arena_thread_freeres+0xfaf>
 496d03:	4c 3b 7e 28          	cmp    0x28(%rsi),%r15
 496d07:	0f 85 1e fd ff ff    	jne    496a2b <arena_thread_freeres+0xd4b>
 496d0d:	49 8b 7f 28          	mov    0x28(%r15),%rdi
 496d11:	4c 3b 7f 20          	cmp    0x20(%rdi),%r15
 496d15:	0f 85 10 fd ff ff    	jne    496a2b <arena_thread_freeres+0xd4b>
 496d1b:	48 83 78 20 00       	cmpq   $0x0,0x20(%rax)
 496d20:	74 1e                	je     496d40 <arena_thread_freeres+0x1060>
 496d22:	48 89 7e 28          	mov    %rdi,0x28(%rsi)
 496d26:	49 8b 47 28          	mov    0x28(%r15),%rax
 496d2a:	48 89 70 20          	mov    %rsi,0x20(%rax)
 496d2e:	e9 d4 fa ff ff       	jmpq   496807 <arena_thread_freeres+0xb27>
 496d33:	48 89 52 28          	mov    %rdx,0x28(%rdx)
 496d37:	48 89 52 20          	mov    %rdx,0x20(%rdx)
 496d3b:	e9 9a f6 ff ff       	jmpq   4963da <arena_thread_freeres+0x6fa>
 496d40:	49 39 f7             	cmp    %rsi,%r15
 496d43:	74 1d                	je     496d62 <arena_thread_freeres+0x1082>
 496d45:	48 89 70 20          	mov    %rsi,0x20(%rax)
 496d49:	49 8b 77 20          	mov    0x20(%r15),%rsi
 496d4d:	48 89 78 28          	mov    %rdi,0x28(%rax)
 496d51:	48 89 46 28          	mov    %rax,0x28(%rsi)
 496d55:	49 8b 77 28          	mov    0x28(%r15),%rsi
 496d59:	48 89 46 20          	mov    %rax,0x20(%rsi)
 496d5d:	e9 a5 fa ff ff       	jmpq   496807 <arena_thread_freeres+0xb27>
 496d62:	48 89 40 28          	mov    %rax,0x28(%rax)
 496d66:	48 89 40 20          	mov    %rax,0x20(%rax)
 496d6a:	e9 98 fa ff ff       	jmpq   496807 <arena_thread_freeres+0xb27>

Disassembly of section .fini:

0000000000496d70 <_fini>:
```

```
496d70:     48 83 ec 08          sub    $0x8,%rsp
496d74:     48 83 c4 08          add    $0x8,%rsp
496d78:     c3                retq
```