Compiler Optimization Notes

Mac Radigan

Compiler Optimization

Variable Declaration

declared in base scope

```
// variable_stack_1.cc

  #include <stdio.h>

  int main(void) {
    int value;
    for(int k=0; k<10; ++k)
    {
      value = k;
      fprintf(stdout, "value: %d\n", value);
    }
    return 0;
  }

// *EOF*
```

```

./bin/variable_stack_1:     file format elf64-x86-64


Disassembly of section .init:

00000000004003b8 <_init>:
  4003b8:   48 83 ec 08             sub    $0x8,%rsp
  4003bc:   48 8b 05 1d 05 20 00    mov    0x20051d(%rip),%rax        # 6008e0 <_DYNAMIC+0x1d0>
  4003c3:   48 85 c0                test   %rax,%rax
  4003c6:   74 05                   je     4003cd <_init+0x15>
  4003c8:   e8 43 00 00 00          callq  400410 <fprintf@plt+0x10>
  4003cd:   48 83 c4 08             add    $0x8,%rsp
  4003d1:   c3                      retq
```

```

Disassembly of section .plt:

00000000004003e0 <__libc_start_main@plt-0x10>:
  4003e0:   ff 35 0a 05 20 00       pushq  0x20050a(%rip)        # 6008f0 <_GLOBAL_OFFSET_TABLE_+0x8>
  4003e6:   ff 25 0c 05 20 00       jmpq   *0x20050c(%rip)        # 6008f8 <_GLOBAL_OFFSET_TABLE_+0x10>
  4003ec:   0f 1f 40 00             nopl   0x0(%rax)

00000000004003f0 <__libc_start_main@plt>:
  4003f0:   ff 25 0a 05 20 00       jmpq   *0x20050a(%rip)        # 600900 <_GLOBAL_OFFSET_TABLE_+0x18>
  4003f6:   68 00 00 00 00          pushq  $0x0
  4003fb:   e9 e0 ff ff ff          jmpq   4003e0 <_init+0x28>

0000000000400400 <fprintf@plt>:
  400400:   ff 25 02 05 20 00       jmpq   *0x200502(%rip)        # 600908 <_GLOBAL_OFFSET_TABLE_+0x20>
  400406:   68 01 00 00 00          pushq  $0x1
  40040b:   e9 d0 ff ff ff          jmpq   4003e0 <_init+0x28>

Disassembly of section .plt.got:

0000000000400410 <.plt.got>:
  400410:   ff 25 ca 04 20 00       jmpq   *0x2004ca(%rip)        # 6008e0 <_DYNAMIC+0x1d0>
  400416:   66 90                   xchg   %ax,%ax

Disassembly of section .text:

0000000000400420 <_start>:
  400420:   31 ed                   xor    %ebp,%ebp
  400422:   49 89 d1                mov    %rdx,%r9
  400425:   5e                      pop    %rsi
  400426:   48 89 e2                mov    %rsp,%rdx
  400429:   48 83 e4 f0             and    $0xfffffffffffffff0,%rsp
  40042d:   50                      push   %rax
  40042e:   54                      push   %rsp
  40042f:   49 c7 c0 b0 05 40 00    mov    $0x4005b0,%r8
  400436:   48 c7 c1 40 05 40 00    mov    $0x400540,%rcx
  40043d:   48 c7 c7 f7 04 40 00    mov    $0x4004f7,%rdi
  400444:   e8 a7 ff ff ff          callq  4003f0 <__libc_start_main@plt>
  400449:   f4                      hlt
  40044a:   66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)

0000000000400450 <deregister_tm_clones>:
  400450:   55                      push   %rbp
  400451:   b8 20 09 60 00          mov    $0x600920,%eax
  400456:   48 3d 20 09 60 00       cmp    $0x600920,%rax
  40045c:   48 89 e5                mov    %rsp,%rbp
  40045f:   74 17                   je     400478 <deregister_tm_clones+0x28>
  400461:   b8 00 00 00 00          mov    $0x0,%eax
  400466:   48 85 c0                test   %rax,%rax
  400469:   74 0d                   je     400478 <deregister_tm_clones+0x28>
  40046b:   5d                      pop    %rbp
  40046c:   bf 20 09 60 00          mov    $0x600920,%edi
  400471:   ff e0                   jmpq   *%rax
  400473:   0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)
  400478:   5d                      pop    %rbp
  400479:   c3                      retq
  40047a:   66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)
```

```
0000000000400480 <register_tm_clones>:
  400480:   be 20 09 60 00          mov    $0x600920,%esi
  400485:   55                      push   %rbp
  400486:   48 81 ee 20 09 60 00    sub    $0x600920,%rsi
  40048d:   48 89 e5                mov    %rsp,%rbp
  400490:   48 c1 fe 03             sar    $0x3,%rsi
  400494:   48 89 f0                mov    %rsi,%rax
  400497:   48 c1 e8 3f             shr    $0x3f,%rax
  40049b:   48 01 c6                add    %rax,%rsi
  40049e:   48 d1 fe                sar    %rsi
  4004a1:   74 15                   je     4004b8 <register_tm_clones+0x38>
  4004a3:   b8 00 00 00 00          mov    $0x0,%eax
  4004a8:   48 85 c0                test   %rax,%rax
  4004ab:   74 0b                   je     4004b8 <register_tm_clones+0x38>
  4004ad:   5d                      pop    %rbp
  4004ae:   bf 20 09 60 00          mov    $0x600920,%edi
  4004b3:   ff e0                   jmpq   *%rax
  4004b5:   0f 1f 00                nopl   (%rax)
  4004b8:   5d                      pop    %rbp
  4004b9:   c3                      retq
  4004ba:   66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)

00000000004004c0 <__do_global_dtors_aux>:
  4004c0:   80 3d 61 04 20 00 00    cmpb   $0x0,0x200461(%rip)        # 600928 <completed.6981>
  4004c7:   75 17                   jne    4004e0 <__do_global_dtors_aux+0x20>
  4004c9:   55                      push   %rbp
  4004ca:   48 89 e5                mov    %rsp,%rbp
  4004cd:   e8 7e ff ff ff          callq  400450 <deregister_tm_clones>
  4004d2:   c6 05 4f 04 20 00 01    movb   $0x1,0x20044f(%rip)        # 600928 <completed.6981>
  4004d9:   5d                      pop    %rbp
  4004da:   c3                      retq
  4004db:   0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)
  4004e0:   f3 c3                   repz retq
  4004e2:   0f 1f 40 00             nopl   0x0(%rax)
  4004e6:   66 2e 0f 1f 84 00 00    nopw   %cs:0x0(%rax,%rax,1)
  4004ed:   00 00 00

00000000004004f0 <frame_dummy>:
  4004f0:   55                      push   %rbp
  4004f1:   48 89 e5                mov    %rsp,%rbp
  4004f4:   5d                      pop    %rbp
  4004f5:   eb 89                   jmp    400480 <register_tm_clones>

00000000004004f7 <main>:
  4004f7:   55                      push   %rbp
  4004f8:   48 89 e5                mov    %rsp,%rbp
  4004fb:   48 83 ec 10             sub    $0x10,%rsp
  4004ff:   c7 45 fc 00 00 00 00    movl   $0x0,-0x4(%rbp)
  400506:   83 7d fc 09             cmpl   $0x9,-0x4(%rbp)
  40050a:   7f 28                   jg     400534 <main+0x3d>
  40050c:   8b 45 fc                mov    -0x4(%rbp),%eax
  40050f:   89 45 f8                mov    %eax,-0x8(%rbp)
  400512:   48 8b 05 07 04 20 00    mov    0x200407(%rip),%rax        # 600920 <__TMC_END__>
  400519:   8b 55 f8                mov    -0x8(%rbp),%edx
  40051c:   be c4 05 40 00          mov    $0x4005c4,%esi
  400521:   48 89 c7                mov    %rax,%rdi
  400524:   b8 00 00 00 00          mov    $0x0,%eax
  400529:   e8 d2 fe ff ff          callq  400400 <fprintf@plt>
  40052e:   83 45 fc 01             addl   $0x1,-0x4(%rbp)
  400532:   eb d2                   jmp    400506 <main+0xf>
  400534:   b8 00 00 00 00          mov    $0x0,%eax
  400539:   c9                      leaveq
  40053a:   c3                      retq
  40053b:   0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)

0000000000400540 <__libc_csu_init>:
  400540:   41 57                   push   %r15
  400542:   41 56                   push   %r14
  400544:   41 89 ff                mov    %edi,%r15d
  400547:   41 55                   push   %r13
  400549:   41 54                   push   %r12
  40054b:   4c 8d 25 ae 01 20 00    lea    0x2001ae(%rip),%r12        # 600700 <__frame_dummy_init_array_entry>
  400552:   55                      push   %rbp
  400553:   48 8d 2d ae 01 20 00    lea    0x2001ae(%rip),%rbp        # 600708 <__init_array_end>
  40055a:   53                      push   %rbx
  40055b:   49 89 f6                mov    %rsi,%r14
  40055e:   49 89 d5                mov    %rdx,%r13
  400561:   4c 29 e5                sub    %r12,%rbp
  400564:   48 83 ec 08             sub    $0x8,%rsp
  400568:   48 c1 fd 03             sar    $0x3,%rbp
  40056c:   e8 47 fe ff ff          callq  4003b8 <_init>
  400571:   48 85 ed                test   %rbp,%rbp
  400574:   74 20                   je     400596 <__libc_csu_init+0x56>
  400576:   31 db                   xor    %ebx,%ebx
  400578:   0f 1f 84 00 00 00 00    nopl   0x0(%rax,%rax,1)
  40057f:   00
  400580:   4c 89 ea                mov    %r13,%rdx
  400583:   4c 89 f6                mov    %r14,%rsi
  400586:   44 89 ff                mov    %r15d,%edi
  400589:   41 ff 14 dc             callq  *(%r12,%rbx,8)
  40058d:   48 83 c3 01             add    $0x1,%rbx
  400591:   48 39 eb                cmp    %rbp,%rbx
  400594:   75 ea                   jne    400580 <__libc_csu_init+0x40>
  400596:   48 83 c4 08             add    $0x8,%rsp
  40059a:   5b                      pop    %rbx
  40059b:   5d                      pop    %rbp
  40059c:   41 5c                   pop    %r12
  40059e:   41 5d                   pop    %r13
  4005a0:   41 5e                   pop    %r14
  4005a2:   41 5f                   pop    %r15
  4005a4:   c3                      retq
  4005a5:   90                      nop
  4005a6:   66 2e 0f 1f 84 00 00    nopw   %cs:0x0(%rax,%rax,1)
  4005ad:   00 00 00

00000000004005b0 <__libc_csu_fini>:
  4005b0:   f3 c3                   repz retq

Disassembly of section .fini:

00000000004005b4 <_fini>:
  4005b4:   48 83 ec 08             sub    $0x8,%rsp
  4005b8:   48 83 c4 08             add    $0x8,%rsp
  4005bc:   c3                      retq
```

declared within statement block

```
// variable_stack_1.cc

  #include <stdio.h>

  int main(void) {
    for(int k=0; k<10; ++k)
    {
      int value = k;
      fprintf(stdout, "value: %d\n", value);
    }
    return 0;
  }

// *EOF*
```

```

./bin/variable_stack_2:     file format elf64-x86-64


Disassembly of section .init:

00000000004003b8 <_init>:
  4003b8:   48 83 ec 08             sub    $0x8,%rsp
  4003bc:   48 8b 05 1d 05 20 00    mov    0x20051d(%rip),%rax        # 6008e0 <_DYNAMIC+0x1d0>
  4003c3:   48 85 c0                test   %rax,%rax
  4003c6:   74 05                   je     4003cd <_init+0x15>
  4003c8:   e8 43 00 00 00          callq  400410 <fprintf@plt+0x10>
  4003cd:   48 83 c4 08             add    $0x8,%rsp
  4003d1:   c3                      retq

Disassembly of section .plt:

00000000004003e0 <__libc_start_main@plt-0x10>:
  4003e0:   ff 35 0a 05 20 00       pushq  0x20050a(%rip)        # 6008f0 <_GLOBAL_OFFSET_TABLE_+0x8>
  4003e6:   ff 25 0c 05 20 00       jmpq   *0x20050c(%rip)        # 6008f8 <_GLOBAL_OFFSET_TABLE_+0x10>
  4003ec:   0f 1f 40 00             nopl   0x0(%rax)

00000000004003f0 <__libc_start_main@plt>:
  4003f0:   ff 25 0a 05 20 00       jmpq   *0x20050a(%rip)        # 600900 <_GLOBAL_OFFSET_TABLE_+0x18>
  4003f6:   68 00 00 00 00          pushq  $0x0
  4003fb:   e9 e0 ff ff ff          jmpq   4003e0 <_init+0x28>

0000000000400400 <fprintf@plt>:
  400400:   ff 25 02 05 20 00       jmpq   *0x200502(%rip)        # 600908 <_GLOBAL_OFFSET_TABLE_+0x20>
  400406:   68 01 00 00 00          pushq  $0x1
  40040b:   e9 d0 ff ff ff          jmpq   4003e0 <_init+0x28>
```

```

Disassembly of section .plt.got:

0000000000400410 <.plt.got>:
  400410:   ff 25 ca 04 20 00       jmpq   *0x2004ca(%rip)        # 6008e0 <_DYNAMIC+0x1d0>
  400416:   66 90                   xchg   %ax,%ax

Disassembly of section .text:

0000000000400420 <_start>:
  400420:   31 ed                   xor    %ebp,%ebp
  400422:   49 89 d1                mov    %rdx,%r9
  400425:   5e                      pop    %rsi
  400426:   48 89 e2                mov    %rsp,%rdx
  400429:   48 83 e4 f0             and    $0xfffffffffffffff0,%rsp
  40042d:   50                      push   %rax
  40042e:   54                      push   %rsp
  40042f:   49 c7 c0 b0 05 40 00    mov    $0x4005b0,%r8
  400436:   48 c7 c1 40 05 40 00    mov    $0x400540,%rcx
  40043d:   48 c7 c7 f7 04 40 00    mov    $0x4004f7,%rdi
  400444:   e8 a7 ff ff ff          callq  4003f0 <__libc_start_main@plt>
  400449:   f4                      hlt
  40044a:   66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)

0000000000400450 <deregister_tm_clones>:
  400450:   55                      push   %rbp
  400451:   b8 20 09 60 00          mov    $0x600920,%eax
  400456:   48 3d 20 09 60 00       cmp    $0x600920,%rax
  40045c:   48 89 e5                mov    %rsp,%rbp
  40045f:   74 17                   je     400478 <deregister_tm_clones+0x28>
  400461:   b8 00 00 00 00          mov    $0x0,%eax
  400466:   48 85 c0                test   %rax,%rax
  400469:   74 0d                   je     400478 <deregister_tm_clones+0x28>
  40046b:   5d                      pop    %rbp
  40046c:   bf 20 09 60 00          mov    $0x600920,%edi
  400471:   ff e0                   jmpq   *%rax
  400473:   0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)
  400478:   5d                      pop    %rbp
  400479:   c3                      retq
  40047a:   66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)

0000000000400480 <register_tm_clones>:
  400480:   be 20 09 60 00          mov    $0x600920,%esi
  400485:   55                      push   %rbp
  400486:   48 81 ee 20 09 60 00    sub    $0x600920,%rsi
  40048d:   48 89 e5                mov    %rsp,%rbp
  400490:   48 c1 fe 03             sar    $0x3,%rsi
  400494:   48 89 f0                mov    %rsi,%rax
  400497:   48 c1 e8 3f             shr    $0x3f,%rax
  40049b:   48 01 c6                add    %rax,%rsi
  40049e:   48 d1 fe                sar    %rsi
  4004a1:   74 15                   je     4004b8 <register_tm_clones+0x38>
  4004a3:   b8 00 00 00 00          mov    $0x0,%eax
  4004a8:   48 85 c0                test   %rax,%rax
  4004ab:   74 0b                   je     4004b8 <register_tm_clones+0x38>
  4004ad:   5d                      pop    %rbp
  4004ae:   bf 20 09 60 00          mov    $0x600920,%edi
  4004b3:   ff e0                   jmpq   *%rax
  4004b5:   0f 1f 00                nopl   (%rax)
  4004b8:   5d                      pop    %rbp
  4004b9:   c3                      retq
  4004ba:   66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)

00000000004004c0 <__do_global_dtors_aux>:
  4004c0:   80 3d 61 04 20 00 00    cmpb   $0x0,0x200461(%rip)        # 600928 <completed.6981>
  4004c7:   75 17                   jne    4004e0 <__do_global_dtors_aux+0x20>
  4004c9:   55                      push   %rbp
  4004ca:   48 89 e5                mov    %rsp,%rbp
  4004cd:   e8 7e ff ff ff          callq  400450 <deregister_tm_clones>
  4004d2:   c6 05 4f 04 20 00 01    movb   $0x1,0x20044f(%rip)        # 600928 <completed.6981>
  4004d9:   5d                      pop    %rbp
  4004da:   c3                      retq
  4004db:   0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)
  4004e0:   f3 c3                   repz retq
  4004e2:   0f 1f 40 00             nopl   0x0(%rax)
  4004e6:   66 2e 0f 1f 84 00 00    nopw   %cs:0x0(%rax,%rax,1)
  4004ed:   00 00 00

00000000004004f0 <frame_dummy>:
  4004f0:   55                      push   %rbp
  4004f1:   48 89 e5                mov    %rsp,%rbp
  4004f4:   5d                      pop    %rbp
  4004f5:   eb 89                   jmp    400480 <register_tm_clones>

00000000004004f7 <main>:
  4004f7:   55                      push   %rbp
  4004f8:   48 89 e5                mov    %rsp,%rbp
  4004fb:   48 83 ec 10             sub    $0x10,%rsp
  4004ff:   c7 45 fc 00 00 00 00    movl   $0x0,-0x4(%rbp)
  400506:   83 7d fc 09             cmpl   $0x9,-0x4(%rbp)
  40050a:   7f 28                   jg     400534 <main+0x3d>
  40050c:   8b 45 fc                mov    -0x4(%rbp),%eax
  40050f:   89 45 f8                mov    %eax,-0x8(%rbp)
  400512:   48 8b 05 07 04 20 00    mov    0x200407(%rip),%rax        # 600920 <__TMC_END__>
  400519:   8b 55 f8                mov    -0x8(%rbp),%edx
  40051c:   be c4 05 40 00          mov    $0x4005c4,%esi
  400521:   48 89 c7                mov    %rax,%rdi
  400524:   b8 00 00 00 00          mov    $0x0,%eax
  400529:   e8 d2 fe ff ff          callq  400400 <fprintf@plt>
  40052e:   83 45 fc 01             addl   $0x1,-0x4(%rbp)
  400532:   eb d2                   jmp    400506 <main+0xf>
  400534:   b8 00 00 00 00          mov    $0x0,%eax
  400539:   c9                      leaveq
  40053a:   c3                      retq
  40053b:   0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)

0000000000400540 <__libc_csu_init>:
  400540:   41 57                   push   %r15
  400542:   41 56                   push   %r14
  400544:   41 89 ff                mov    %edi,%r15d
  400547:   41 55                   push   %r13
  400549:   41 54                   push   %r12
  40054b:   4c 8d 25 ae 01 20 00    lea    0x2001ae(%rip),%r12        # 600700 <__frame_dummy_init_array_entry>
  400552:   55                      push   %rbp
  400553:   48 8d 2d ae 01 20 00    lea    0x2001ae(%rip),%rbp        # 600708 <__init_array_end>
  40055a:   53                      push   %rbx
  40055b:   49 89 f6                mov    %rsi,%r14
  40055e:   49 89 d5                mov    %rdx,%r13
  400561:   4c 29 e5                sub    %r12,%rbp
  400564:   48 83 ec 08             sub    $0x8,%rsp
  400568:   48 c1 fd 03             sar    $0x3,%rbp
  40056c:   e8 47 fe ff ff          callq  4003b8 <_init>
  400571:   48 85 ed                test   %rbp,%rbp
  400574:   74 20                   je     400596 <__libc_csu_init+0x56>
  400576:   31 db                   xor    %ebx,%ebx
  400578:   0f 1f 84 00 00 00 00    nopl   0x0(%rax,%rax,1)
  40057f:   00
  400580:   4c 89 ea                mov    %r13,%rdx
  400583:   4c 89 f6                mov    %r14,%rsi
  400586:   44 89 ff                mov    %r15d,%edi
  400589:   41 ff 14 dc             callq  *(%r12,%rbx,8)
  40058d:   48 83 c3 01             add    $0x1,%rbx
  400591:   48 39 eb                cmp    %rbp,%rbx
  400594:   75 ea                   jne    400580 <__libc_csu_init+0x40>
  400596:   48 83 c4 08             add    $0x8,%rsp
  40059a:   5b                      pop    %rbx
  40059b:   5d                      pop    %rbp
  40059c:   41 5c                   pop    %r12
  40059e:   41 5d                   pop    %r13
  4005a0:   41 5e                   pop    %r14
  4005a2:   41 5f                   pop    %r15
  4005a4:   c3                      retq
  4005a5:   90                      nop
  4005a6:   66 2e 0f 1f 84 00 00    nopw   %cs:0x0(%rax,%rax,1)
  4005ad:   00 00 00

00000000004005b0 <__libc_csu_fini>:
  4005b0:   f3 c3                   repz retq

Disassembly of section .fini:

00000000004005b4 <_fini>:
  4005b4:   48 83 ec 08             sub    $0x8,%rsp
  4005b8:   48 83 c4 08             add    $0x8,%rsp
  4005bc:   c3                      retq
```

implementation difference (side-by-side diff, no suppression)

```


Disassembly of section .init:                           Disassembly of section .init:

00000000004003b8 <_init>:                           00000000004003b8 <_init>:
  4003b8:   48 83 ec 08             sub    $0x8,%rsp              4003b8:   48 83 ec 08             sub    $0x8,%rsp
  4003bc:   48 8b 05 1d 05 20 00    mov    0x20051d(%rip),%rax            4003bc:   48 8b 05 1d 05 20 00    mov    0x20051d(%rip),%rax
  4003c3:   48 85 c0                test   %rax,%rax              4003c3:   48 85 c0                test   %rax,%rax
  4003c6:   74 05                   je     4003cd <_init+0x15>        4003c6:   74 05                   je     4003cd <_init+0x15>
  4003c8:   e8 43 00 00 00          callq  400410 <fprintf@plt+0x1        4003c8:   e8 43 00 00 00          callq  400410 <fprintf@plt+0x1
  4003cd:   48 83 c4 08             add    $0x8,%rsp              4003cd:   48 83 c4 08             add    $0x8,%rsp
  4003d1:   c3                      retq                      4003d1:   c3                      retq

Disassembly of section .plt:                            Disassembly of section .plt:

```

```
00000000004003e0 <__libc_start_main@plt-0x10>:                   00000000004003e0 <__libc_start_main@plt-0x10>:
  4003e0:   ff 35 0a 05 20 00       pushq  0x20050a(%rip)        #        4003e0:   ff 35 0a 05 20 00       pushq  0x20050a(%rip)         #
  4003e6:   ff 25 0c 05 20 00       jmpq   *0x20050c(%rip)                4003e6:   ff 25 0c 05 20 00       jmpq   *0x20050c(%rip)
  4003ec:   0f 1f 40 00             nopl   0x0(%rax)             4003ec:   0f 1f 40 00             nopl   0x0(%rax)

00000000004003f0 <__libc_start_main@plt>:                   00000000004003f0 <__libc_start_main@plt>:
  4003f0:   ff 25 0a 05 20 00       jmpq   *0x20050a(%rip)                4003f0:   ff 25 0a 05 20 00       jmpq   *0x20050a(%rip)
  4003f6:   68 00 00 00 00          pushq  $0x0               4003f6:   68 00 00 00 00          pushq  $0x0
  4003fb:   e9 e0 ff ff ff          jmpq   4003e0 <_init+0x28>        4003fb:   e9 e0 ff ff ff          jmpq   4003e0 <_init+0x28>

0000000000400400 <fprintf@plt>:                         0000000000400400 <fprintf@plt>:
  400400:   ff 25 02 05 20 00       jmpq   *0x200502(%rip)                400400:   ff 25 02 05 20 00       jmpq   *0x200502(%rip)
  400406:   68 01 00 00 00          pushq  $0x1               400406:   68 01 00 00 00          pushq  $0x1
  40040b:   e9 d0 ff ff ff          jmpq   4003e0 <_init+0x28>        40040b:   e9 d0 ff ff ff          jmpq   4003e0 <_init+0x28>

Disassembly of section .plt.got:                        Disassembly of section .plt.got:

0000000000400410 <.plt.got>:                            0000000000400410 <.plt.got>:
  400410:   ff 25 ca 04 20 00       jmpq   *0x2004ca(%rip)                400410:   ff 25 ca 04 20 00       jmpq   *0x2004ca(%rip)
  400416:   66 90                   xchg   %ax,%ax                400416:   66 90                   xchg   %ax,%ax

Disassembly of section .text:                           Disassembly of section .text:

0000000000400420 <_start>:                          0000000000400420 <_start>:
  400420:   31 ed                   xor    %ebp,%ebp              400420:   31 ed                   xor    %ebp,%ebp
  400422:   49 89 d1                mov    %rdx,%r9               400422:   49 89 d1                mov    %rdx,%r9
  400425:   5e                      pop    %rsi               400425:   5e                      pop    %rsi
  400426:   48 89 e2                mov    %rsp,%rdx              400426:   48 89 e2                mov    %rsp,%rdx
  400429:   48 83 e4 f0             and    $0xfffffffffffffff0,%rs        400429:   48 83 e4 f0             and    $0xfffffffffffffff0,%rs
  40042d:   50                      push   %rax               40042d:   50                      push   %rax
  40042e:   54                      push   %rsp               40042e:   54                      push   %rsp
  40042f:   49 c7 c0 b0 05 40 00    mov    $0x4005b0,%r8               40042f:   49 c7 c0 b0 05 40 00    mov    $0x4005b0,%r8
  400436:   48 c7 c1 40 05 40 00    mov    $0x400540,%rcx             400436:   48 c7 c1 40 05 40 00    mov    $0x400540,%rcx
  40043d:   48 c7 c7 f7 04 40 00    mov    $0x4004f7,%rdi             40043d:   48 c7 c7 f7 04 40 00    mov    $0x4004f7,%rdi
  400444:   e8 a7 ff ff ff          callq  4003f0 <__libc_start_ma        400444:   e8 a7 ff ff ff          callq  4003f0 <__libc_start_ma
  400449:   f4                      hlt                       400449:   f4                      hlt
  40044a:   66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)           40044a:   66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)

0000000000400450 <deregister_tm_clones>:                    0000000000400450 <deregister_tm_clones>:
  400450:   55                      push   %rbp               400450:   55                      push   %rbp
  400451:   b8 20 09 60 00          mov    $0x600920,%eax             400451:   b8 20 09 60 00          mov    $0x600920,%eax
  400456:   48 3d 20 09 60 00       cmp    $0x600920,%rax             400456:   48 3d 20 09 60 00       cmp    $0x600920,%rax
  40045c:   48 89 e5                mov    %rsp,%rbp              40045c:   48 89 e5                mov    %rsp,%rbp
  40045f:   74 17                   je     400478 <deregister_tm_c        40045f:   74 17                   je     400478 <deregister_tm_c
  400461:   b8 00 00 00 00          mov    $0x0,%eax              400461:   b8 00 00 00 00          mov    $0x0,%eax
  400466:   48 85 c0                test   %rax,%rax              400466:   48 85 c0                test   %rax,%rax
  400469:   74 0d                   je     400478 <deregister_tm_c        400469:   74 0d                   je     400478 <deregister_tm_c
  40046b:   5d                      pop    %rbp               40046b:   5d                      pop    %rbp
  40046c:   bf 20 09 60 00          mov    $0x600920,%edi             40046c:   bf 20 09 60 00          mov    $0x600920,%edi
  400471:   ff e0                   jmpq   *%rax                  400471:   ff e0                   jmpq   *%rax
  400473:   0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)           400473:   0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)
  400478:   5d                      pop    %rbp               400478:   5d                      pop    %rbp
  400479:   c3                      retq                      400479:   c3                      retq
  40047a:   66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)           40047a:   66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)

0000000000400480 <register_tm_clones>:                      0000000000400480 <register_tm_clones>:
  400480:   be 20 09 60 00          mov    $0x600920,%esi             400480:   be 20 09 60 00          mov    $0x600920,%esi
  400485:   55                      push   %rbp               400485:   55                      push   %rbp
  400486:   48 81 ee 20 09 60 00    sub    $0x600920,%rsi             400486:   48 81 ee 20 09 60 00    sub    $0x600920,%rsi
  40048d:   48 89 e5                mov    %rsp,%rbp              40048d:   48 89 e5                mov    %rsp,%rbp
  400490:   48 c1 fe 03             sar    $0x3,%rsi              400490:   48 c1 fe 03             sar    $0x3,%rsi
  400494:   48 89 f0                mov    %rsi,%rax              400494:   48 89 f0                mov    %rsi,%rax
  400497:   48 c1 e8 3f             shr    $0x3f,%rax             400497:   48 c1 e8 3f             shr    $0x3f,%rax
  40049b:   48 01 c6                add    %rax,%rsi              40049b:   48 01 c6                add    %rax,%rsi
  40049e:   48 d1 fe                sar    %rsi               40049e:   48 d1 fe                sar    %rsi
  4004a1:   74 15                   je     4004b8 <register_tm_clo        4004a1:   74 15                   je     4004b8 <register_tm_clo
  4004a3:   b8 00 00 00 00          mov    $0x0,%eax              4004a3:   b8 00 00 00 00          mov    $0x0,%eax
  4004a8:   48 85 c0                test   %rax,%rax              4004a8:   48 85 c0                test   %rax,%rax
  4004ab:   74 0b                   je     4004b8 <register_tm_clo        4004ab:   74 0b                   je     4004b8 <register_tm_clo
  4004ad:   5d                      pop    %rbp               4004ad:   5d                      pop    %rbp
  4004ae:   bf 20 09 60 00          mov    $0x600920,%edi             4004ae:   bf 20 09 60 00          mov    $0x600920,%edi
  4004b3:   ff e0                   jmpq   *%rax                  4004b3:   ff e0                   jmpq   *%rax
  4004b5:   0f 1f 00                nopl   (%rax)                 4004b5:   0f 1f 00                nopl   (%rax)
  4004b8:   5d                      pop    %rbp               4004b8:   5d                      pop    %rbp
  4004b9:   c3                      retq                      4004b9:   c3                      retq
  4004ba:   66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)           4004ba:   66 0f 1f 44 00 00       nopw   0x0(%rax,%rax,1)

00000000004004c0 <__do_global_dtors_aux>:                   00000000004004c0 <__do_global_dtors_aux>:
  4004c0:   80 3d 61 04 20 00 00    cmpb   $0x0,0x200461(%rip)            4004c0:   80 3d 61 04 20 00 00    cmpb   $0x0,0x200461(%rip)
  4004c7:   75 17                   jne    4004e0 <__do_global_dto        4004c7:   75 17                   jne    4004e0 <__do_global_dto
  4004c9:   55                      push   %rbp               4004c9:   55                      push   %rbp
  4004ca:   48 89 e5                mov    %rsp,%rbp              4004ca:   48 89 e5                mov    %rsp,%rbp
  4004cd:   e8 7e ff ff ff          callq  400450 <deregister_tm_c        4004cd:   e8 7e ff ff ff          callq  400450 <deregister_tm_c
  4004d2:   c6 05 4f 04 20 00 01    movb   $0x1,0x20044f(%rip)            4004d2:   c6 05 4f 04 20 00 01    movb   $0x1,0x20044f(%rip)
  4004d9:   5d                      pop    %rbp               4004d9:   5d                      pop    %rbp
  4004da:   c3                      retq                      4004da:   c3                      retq
  4004db:   0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)           4004db:   0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)
  4004e0:   f3 c3                   repz retq                 4004e0:   f3 c3                   repz retq
  4004e2:   0f 1f 40 00             nopl   0x0(%rax)             4004e2:   0f 1f 40 00             nopl   0x0(%rax)
  4004e6:   66 2e 0f 1f 84 00 00    nopw   %cs:0x0(%rax,%rax,1)       4004e6:   66 2e 0f 1f 84 00 00    nopw   %cs:0x0(%rax,%rax,1)
  4004ed:   00 00 00                              4004ed:   00 00 00

00000000004004f0 <frame_dummy>:                         00000000004004f0 <frame_dummy>:
  4004f0:   55                      push   %rbp               4004f0:   55                      push   %rbp
  4004f1:   48 89 e5                mov    %rsp,%rbp              4004f1:   48 89 e5                mov    %rsp,%rbp
  4004f4:   5d                      pop    %rbp               4004f4:   5d                      pop    %rbp
  4004f5:   eb 89                   jmp    400480 <register_tm_clo        4004f5:   eb 89                   jmp    400480 <register_tm_clo

00000000004004f7 <main>:                            00000000004004f7 <main>:
  4004f7:   55                      push   %rbp               4004f7:   55                      push   %rbp
  4004f8:   48 89 e5                mov    %rsp,%rbp              4004f8:   48 89 e5                mov    %rsp,%rbp
  4004fb:   48 83 ec 10             sub    $0x10,%rsp             4004fb:   48 83 ec 10             sub    $0x10,%rsp
  4004ff:   c7 45 fc 00 00 00 00    movl   $0x0,-0x4(%rbp)            4004ff:   c7 45 fc 00 00 00 00    movl   $0x0,-0x4(%rbp)
  400506:   83 7d fc 09             cmpl   $0x9,-0x4(%rbp)            400506:   83 7d fc 09             cmpl   $0x9,-0x4(%rbp)
  40050a:   7f 28                   jg     400534 <main+0x3d>         40050a:   7f 28                   jg     400534 <main+0x3d>
  40050c:   8b 45 fc                mov    -0x4(%rbp),%eax            40050c:   8b 45 fc                mov    -0x4(%rbp),%eax
  40050f:   89 45 f8                mov    %eax,-0x8(%rbp)            40050f:   89 45 f8                mov    %eax,-0x8(%rbp)
  400512:   48 8b 05 07 04 20 00    mov    0x200407(%rip),%rax            400512:   48 8b 05 07 04 20 00    mov    0x200407(%rip),%rax
  400519:   8b 55 f8                mov    -0x8(%rbp),%edx            400519:   8b 55 f8                mov    -0x8(%rbp),%edx
  40051c:   be c4 05 40 00          mov    $0x4005c4,%esi             40051c:   be c4 05 40 00          mov    $0x4005c4,%esi
  400521:   48 89 c7                mov    %rax,%rdi              400521:   48 89 c7                mov    %rax,%rdi
  400524:   b8 00 00 00 00          mov    $0x0,%eax              400524:   b8 00 00 00 00          mov    $0x0,%eax
  400529:   e8 d2 fe ff ff          callq  400400 <fprintf@plt>       400529:   e8 d2 fe ff ff          callq  400400 <fprintf@plt>
  40052e:   83 45 fc 01             addl   $0x1,-0x4(%rbp)            40052e:   83 45 fc 01             addl   $0x1,-0x4(%rbp)
  400532:   eb d2                   jmp    400506 <main+0xf>          400532:   eb d2                   jmp    400506 <main+0xf>
  400534:   b8 00 00 00 00          mov    $0x0,%eax              400534:   b8 00 00 00 00          mov    $0x0,%eax
  400539:   c9                      leaveq                    400539:   c9                      leaveq
  40053a:   c3                      retq                      40053a:   c3                      retq
  40053b:   0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)           40053b:   0f 1f 44 00 00          nopl   0x0(%rax,%rax,1)

0000000000400540 <__libc_csu_init>:                     0000000000400540 <__libc_csu_init>:
  400540:   41 57                   push   %r15               400540:   41 57                   push   %r15
  400542:   41 56                   push   %r14               400542:   41 56                   push   %r14
  400544:   41 89 ff                mov    %edi,%r15d             400544:   41 89 ff                mov    %edi,%r15d
  400547:   41 55                   push   %r13               400547:   41 55                   push   %r13
  400549:   41 54                   push   %r12               400549:   41 54                   push   %r12
  40054b:   4c 8d 25 ae 01 20 00    lea    0x2001ae(%rip),%r12            40054b:   4c 8d 25 ae 01 20 00    lea    0x2001ae(%rip),%r12
  400552:   55                      push   %rbp               400552:   55                      push   %rbp
  400553:   48 8d 2d ae 01 20 00    lea    0x2001ae(%rip),%rbp            400553:   48 8d 2d ae 01 20 00    lea    0x2001ae(%rip),%rbp
  40055a:   53                      push   %rbx               40055a:   53                      push   %rbx
  40055b:   49 89 f6                mov    %rsi,%r14              40055b:   49 89 f6                mov    %rsi,%r14
  40055e:   49 89 d5                mov    %rdx,%r13              40055e:   49 89 d5                mov    %rdx,%r13
  400561:   4c 29 e5                sub    %r12,%rbp              400561:   4c 29 e5                sub    %r12,%rbp
  400564:   48 83 ec 08             sub    $0x8,%rsp              400564:   48 83 ec 08             sub    $0x8,%rsp
  400568:   48 c1 fd 03             sar    $0x3,%rbp              400568:   48 c1 fd 03             sar    $0x3,%rbp
  40056c:   e8 47 fe ff ff          callq  4003b8 <_init>             40056c:   e8 47 fe ff ff          callq  4003b8 <_init>
  400571:   48 85 ed                test   %rbp,%rbp              400571:   48 85 ed                test   %rbp,%rbp
  400574:   74 20                   je     400596 <__libc_csu_init        400574:   74 20                   je     400596 <__libc_csu_init
  400576:   31 db                   xor    %ebx,%ebx              400576:   31 db                   xor    %ebx,%ebx
  400578:   0f 1f 84 00 00 00 00    nopl   0x0(%rax,%rax,1)           400578:   0f 1f 84 00 00 00 00    nopl   0x0(%rax,%rax,1)
  40057f:   00                          40057f:   00
  400580:   4c 89 ea                mov    %r13,%rdx              400580:   4c 89 ea                mov    %r13,%rdx
  400583:   4c 89 f6                mov    %r14,%rsi              400583:   4c 89 f6                mov    %r14,%rsi
  400586:   44 89 ff                mov    %r15d,%edi             400586:   44 89 ff                mov    %r15d,%edi
  400589:   41 ff 14 dc             callq  *(%r12,%rbx,8)             400589:   41 ff 14 dc             callq  *(%r12,%rbx,8)
  40058d:   48 83 c3 01             add    $0x1,%rbx              40058d:   48 83 c3 01             add    $0x1,%rbx
  400591:   48 39 eb                cmp    %rbp,%rbx              400591:   48 39 eb                cmp    %rbp,%rbx
  400594:   75 ea                   jne    400580 <__libc_csu_init        400594:   75 ea                   jne    400580 <__libc_csu_init
  400596:   48 83 c4 08             add    $0x8,%rsp              400596:   48 83 c4 08             add    $0x8,%rsp
  40059a:   5b                      pop    %rbx               40059a:   5b                      pop    %rbx
  40059b:   5d                      pop    %rbp               40059b:   5d                      pop    %rbp
  40059c:   41 5c                   pop    %r12               40059c:   41 5c                   pop    %r12
  40059e:   41 5d                   pop    %r13               40059e:   41 5d                   pop    %r13
  4005a0:   41 5e                   pop    %r14               4005a0:   41 5e                   pop    %r14
  4005a2:   41 5f                   pop    %r15               4005a2:   41 5f                   pop    %r15
  4005a4:   c3                      retq                      4005a4:   c3                      retq
  4005a5:   90                      nop                   4005a5:   90                      nop
  4005a6:   66 2e 0f 1f 84 00 00    nopw   %cs:0x0(%rax,%rax,1)       4005a6:   66 2e 0f 1f 84 00 00    nopw   %cs:0x0(%rax,%rax,1)
  4005ad:   00 00 00                              4005ad:   00 00 00

00000000004005b0 <__libc_csu_fini>:                     00000000004005b0 <__libc_csu_fini>:
  4005b0:   f3 c3                   repz retq                 4005b0:   f3 c3                   repz retq

Disassembly of section .fini:                           Disassembly of section .fini:

00000000004005b4 <_fini>:                           00000000004005b4 <_fini>:
  4005b4:   48 83 ec 08             sub    $0x8,%rsp              4005b4:   48 83 ec 08             sub    $0x8,%rsp
  4005b8:   48 83 c4 08             add    $0x8,%rsp              4005b8:   48 83 c4 08             add    $0x8,%rsp
  4005bc:   c3                      retq                      4005bc:   c3                      retq

== No Difference ==
```

Memory Initialization

declared in .BSS segment

```
// segments_1.cc

  #include <stdio.h>
  #include <string.h>

  #define N (1024)

  int value[N]; // .bss

  int main(void) {
    memset(&value, 'x', N);
    fprintf(stdout, "value: %d\n", value[0]);
    return 0;
  }

// *EOF*
```

```

./bin/segments_1:     file format elf64-x86-64
./bin/segments_1
architecture: i386:x86-64, flags 0x00000112:
EXEC_P, HAS_SYMS, D_PAGED
start address 0x0000000000400470

Program Header:
    PHDR off    0x0000000000000040 vaddr 0x0000000000400040 paddr 0x0000000000400040 align 2**3
         filesz 0x00000000000001c0 memsz 0x00000000000001c0 flags r-x
  INTERP off    0x0000000000000200 vaddr 0x0000000000400200 paddr 0x0000000000400200 align 2**0
         filesz 0x000000000000001c memsz 0x000000000000001c flags r--
    LOAD off    0x0000000000000000 vaddr 0x0000000000400000 paddr 0x0000000000400000 align 2**21
         filesz 0x000000000000074c memsz 0x000000000000074c flags r-x
    LOAD off    0x0000000000000750 vaddr 0x0000000000600750 paddr 0x0000000000600750 align 2**21
         filesz 0x0000000000000228 memsz 0x0000000000001250 flags rw-
 DYNAMIC off    0x0000000000000760 vaddr 0x0000000000600760 paddr 0x0000000000600760 align 2**3
         filesz 0x00000000000001d0 memsz 0x00000000000001d0 flags rw-
    NOTE off    0x000000000000021c vaddr 0x000000000040021c paddr 0x000000000040021c align 2**2
         filesz 0x0000000000000020 memsz 0x0000000000000020 flags r--
EH_FRAME off    0x0000000000000620 vaddr 0x0000000000400620 paddr 0x0000000000400620 align 2**2
         filesz 0x0000000000000034 memsz 0x0000000000000034 flags r--
   STACK off    0x0000000000000000 vaddr 0x0000000000000000 paddr 0x0000000000000000 align 2**4
         filesz 0x0000000000000000 memsz 0x0000000000000000 flags rw-

Dynamic Section:
  NEEDED               libc.so.6
  INIT                 0x00000000004003f8
  FINI                 0x0000000000400604
  INIT_ARRAY           0x0000000000600750
  INIT_ARRAYSZ         0x0000000000000008
  FINI_ARRAY           0x0000000000600758
  FINI_ARRAYSZ         0x0000000000000008
  HASH                 0x0000000000400240
  STRTAB               0x0000000000400300
  SYMTAB               0x0000000000400270
  STRSZ                0x000000000000004e
  SYMENT               0x0000000000000018
  DEBUG                0x0000000000000000
  PLTGOT               0x0000000000600938
  PLTRELSZ             0x0000000000000048
  PLTREL               0x0000000000000007
  JMPREL               0x00000000004003b0
  RELA                 0x0000000000400380
  RELASZ               0x0000000000000030
  RELAENT              0x0000000000000018
  VERNEED              0x0000000000400360
  VERNEEDNUM           0x0000000000000001
  VERSYM               0x000000000040034e

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

Sections:
Idx Name          Size      VMA               LMA               File off  Algn
  0 .interp       0000001c  0000000000400200  0000000000400200  00000200  2**0
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  1 .note.ABI-tag 00000020  000000000040021c  000000000040021c  0000021c  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  2 .hash         0000002c  0000000000400240  0000000000400240  00000240  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  3 .dynsym       00000090  0000000000400270  0000000000400270  00000270  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  4 .dynstr       0000004e  0000000000400300  0000000000400300  00000300  2**0
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  5 .gnu.version  0000000c  000000000040034e  000000000040034e  0000034e  2**1
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  6 .gnu.version_r 00000020  0000000000400360  0000000000400360  00000360  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  7 .rela.dyn     00000030  0000000000400380  0000000000400380  00000380  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  8 .rela.plt     00000048  00000000004003b0  00000000004003b0  000003b0  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  9 .init         0000001a  00000000004003f8  00000000004003f8  000003f8  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
 10 .plt          00000040  0000000000400420  0000000000400420  00000420  2**4
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
 11 .plt.got      00000008  0000000000400460  0000000000400460  00000460  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
 12 .text         00000192  0000000000400470  0000000000400470  00000470  2**4
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
 13 .fini         00000009  0000000000400604  0000000000400604  00000604  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
 14 .rodata       0000000f  0000000000400610  0000000000400610  00000610  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
 15 .eh_frame_hdr 00000034  0000000000400620  0000000000400620  00000620  2**2
```

```
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
 16 .eh_frame     000000f4  0000000000400658  0000000000400658  00000658  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
 17 .init_array   00000008  0000000000600750  0000000000600750  00000750  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 18 .fini_array   00000008  0000000000600758  0000000000600758  00000758  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 19 .dynamic      000001d0  0000000000600760  0000000000600760  00000760  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 20 .got          00000008  0000000000600930  0000000000600930  00000930  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 21 .got.plt      00000030  0000000000600938  0000000000600938  00000938  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 22 .data         00000010  0000000000600968  0000000000600968  00000968  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 23 .bss          00001020  0000000000600980  0000000000600980  00000978  2**5
                  ALLOC
 24 .comment      00000029  0000000000000000  0000000000000000  00000978  2**0
                  CONTENTS, READONLY
 25 .debug_aranges 00000030  0000000000000000  0000000000000000  000009a1  2**0
                  CONTENTS, READONLY, DEBUGGING
 26 .debug_info   00000351  0000000000000000  0000000000000000  000009d1  2**0
                  CONTENTS, READONLY, DEBUGGING
 27 .debug_abbrev 000000fa  0000000000000000  0000000000000000  00000d22  2**0
                  CONTENTS, READONLY, DEBUGGING
 28 .debug_line   000000f0  0000000000000000  0000000000000000  00000e1c  2**0
                  CONTENTS, READONLY, DEBUGGING
 29 .debug_str    000002b8  0000000000000000  0000000000000000  00000f0c  2**0
                  CONTENTS, READONLY, DEBUGGING
SYMBOL TABLE:
0000000000400200 l    d  .interp    0000000000000000              .interp
000000000040021c l    d  .note.ABI-tag  0000000000000000              .note.ABI-tag
0000000000400240 l    d  .hash  0000000000000000              .hash
0000000000400270 l    d  .dynsym    0000000000000000              .dynsym
0000000000400300 l    d  .dynstr    0000000000000000              .dynstr
000000000040034e l    d  .gnu.version   0000000000000000              .gnu.version
0000000000400360 l    d  .gnu.version_r 0000000000000000              .gnu.version_r
0000000000400380 l    d  .rela.dyn  0000000000000000              .rela.dyn
00000000004003b0 l    d  .rela.plt  0000000000000000              .rela.plt
00000000004003f8 l    d  .init  0000000000000000              .init
0000000000400420 l    d  .plt   0000000000000000              .plt
0000000000400460 l    d  .plt.got   0000000000000000              .plt.got
0000000000400470 l    d  .text  0000000000000000              .text
0000000000400604 l    d  .fini  0000000000000000              .fini
0000000000400610 l    d  .rodata    0000000000000000              .rodata
0000000000400620 l    d  .eh_frame_hdr  0000000000000000              .eh_frame_hdr
0000000000400658 l    d  .eh_frame  0000000000000000              .eh_frame
0000000000600750 l    d  .init_array    0000000000000000              .init_array
0000000000600758 l    d  .fini_array    0000000000000000              .fini_array
0000000000600760 l    d  .dynamic   0000000000000000              .dynamic
0000000000600930 l    d  .got   0000000000000000              .got
0000000000600938 l    d  .got.plt   0000000000000000              .got.plt
0000000000600968 l    d  .data  0000000000000000              .data
0000000000600980 l    d  .bss   0000000000000000              .bss
0000000000000000 l    d  .comment   0000000000000000              .comment
0000000000000000 l    d  .debug_aranges 0000000000000000              .debug_aranges
0000000000000000 l    d  .debug_info    0000000000000000              .debug_info
```

```
0000000000000000 l    d  .debug_abbrev  0000000000000000              .debug_abbrev
0000000000000000 l    d  .debug_line    0000000000000000              .debug_line
0000000000000000 l    d  .debug_str 0000000000000000              .debug_str
0000000000000000 l    df *ABS*  0000000000000000              crtstuff.c
00000000004004a0 l     F .text  0000000000000000              deregister_tm_clones
00000000004004d0 l     F .text  0000000000000000              register_tm_clones
0000000000400510 l     F .text  0000000000000000              __do_global_dtors_aux
0000000000600988 l     O .bss   0000000000000001              completed.6981
0000000000600758 l     O .fini_array    0000000000000000              __do_global_dtors_aux_fini_array_entry
0000000000400540 l     F .text  0000000000000000              frame_dummy
0000000000600750 l     O .init_array    0000000000000000              __frame_dummy_init_array_entry
0000000000000000 l    df *ABS*  0000000000000000              segments_1.cc
0000000000000000 l    df *ABS*  0000000000000000              crtstuff.c
0000000000400748 l     O .eh_frame  0000000000000000              __FRAME_END__
0000000000000000 l    df *ABS*  0000000000000000
0000000000600758 l       .init_array    0000000000000000              __init_array_end
0000000000600760 l     O .dynamic   0000000000000000              _DYNAMIC
0000000000600750 l       .init_array    0000000000000000              __init_array_start
0000000000400620 l       .eh_frame_hdr  0000000000000000              __GNU_EH_FRAME_HDR
0000000000600938 l     O .got.plt   0000000000000000              _GLOBAL_OFFSET_TABLE_
0000000000400600 g     F .text  0000000000000002              __libc_csu_fini
0000000000000000  w      *UND*  0000000000000000              _ITM_deregisterTMCloneTable
0000000000600980 g     O .bss   0000000000000008              stdout@@GLIBC_2.2.5
0000000000600968  w      .data  0000000000000000              data_start
0000000000600978 g       .data  0000000000000000              _edata
0000000000400604 g     F .fini  0000000000000000              _fini
0000000000000000       F *UND*  0000000000000000              memset@@GLIBC_2.2.5
0000000000000000       F *UND*  0000000000000000              __libc_start_main@@GLIBC_2.2.5
0000000000600968 g       .data  0000000000000000              __data_start
0000000000000000       F *UND*  0000000000000000              fprintf@@GLIBC_2.2.5
0000000000000000  w      *UND*  0000000000000000              __gmon_start__
00000000006009a0 g     O .bss   0000000000001000              value
0000000000600970 g     O .data  0000000000000000              .hidden __dso_handle
0000000000400610 g     O .rodata    0000000000000004              _IO_stdin_used
0000000000400590 g     F .text  0000000000000065              __libc_csu_init
00000000006019a0 g       .bss   0000000000000000              _end
0000000000400470 g     F .text  000000000000002a              _start
0000000000600978 g       .bss   0000000000000000              __bss_start
0000000000400547 g     F .text  000000000000003e              main
0000000000600978 g     O .data  0000000000000000              .hidden __TMC_END__
0000000000000000  w      *UND*  0000000000000000              _ITM_registerTMCloneTable
00000000004003f8 g     F .init  0000000000000000              _init

```

declared in .DATA segment

```
// segments_2.cc

  #include <stdio.h>

  #define N (1024)

  int value[N]; // .data

  int main(void) {
    value[0] = 0xdeadbeef;
    value[N] = 0xdeadbeef;
    fprintf(stdout, "value: %d\n", value[0]);
    return 0;
  }

// *EOF*
```

```

./bin/segments_2:     file format elf64-x86-64
./bin/segments_2
architecture: i386:x86-64, flags 0x00000112:
EXEC_P, HAS_SYMS, D_PAGED
start address 0x0000000000400420

Program Header:
    PHDR off    0x0000000000000040 vaddr 0x0000000000400040 paddr 0x0000000000400040 align 2**3
         filesz 0x00000000000001c0 memsz 0x00000000000001c0 flags r-x
  INTERP off    0x0000000000000200 vaddr 0x0000000000400200 paddr 0x0000000000400200 align 2**0
         filesz 0x000000000000001c memsz 0x000000000000001c flags r--
    LOAD off    0x0000000000000000 vaddr 0x0000000000400000 paddr 0x0000000000400000 align 2**21
         filesz 0x00000000000006ec memsz 0x00000000000006ec flags r-x
    LOAD off    0x00000000000006f0 vaddr 0x00000000006006f0 paddr 0x00000000006006f0 align 2**21
         filesz 0x0000000000000220 memsz 0x0000000000001250 flags rw-
 DYNAMIC off    0x0000000000000700 vaddr 0x0000000000600700 paddr 0x0000000000600700 align 2**3
         filesz 0x00000000000001d0 memsz 0x00000000000001d0 flags rw-
    NOTE off    0x000000000000021c vaddr 0x000000000040021c paddr 0x000000000040021c align 2**2
         filesz 0x0000000000000020 memsz 0x0000000000000020 flags r--
EH_FRAME off    0x00000000000005c0 vaddr 0x00000000004005c0 paddr 0x00000000004005c0 align 2**2
         filesz 0x0000000000000034 memsz 0x0000000000000034 flags r--
   STACK off    0x0000000000000000 vaddr 0x0000000000000000 paddr 0x0000000000000000 align 2**4
         filesz 0x0000000000000000 memsz 0x0000000000000000 flags rw-

Dynamic Section:
  NEEDED               libc.so.6
  INIT                 0x00000000004003b8
  FINI                 0x00000000004005a4
  INIT_ARRAY           0x00000000006006f0
  INIT_ARRAYSZ         0x0000000000000008
  FINI_ARRAY           0x00000000006006f8
  FINI_ARRAYSZ         0x0000000000000008
  HASH                 0x0000000000400240
  STRTAB               0x00000000004002e0
  SYMTAB               0x0000000000400268
  STRSZ                0x0000000000000047
  SYMENT               0x0000000000000018
  DEBUG                0x0000000000000000
  PLTGOT               0x00000000006008d8
  PLTRELSZ             0x0000000000000030
  PLTREL               0x0000000000000007
  JMPREL               0x0000000000400388
  RELA                 0x0000000000400358
  RELASZ               0x0000000000000030
  RELAENT              0x0000000000000018
  VERNEED              0x0000000000400338
  VERNEEDNUM           0x0000000000000001
  VERSYM               0x0000000000400328

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

Sections:
Idx Name           Size      VMA               LMA               File off  Algn
  0 .interp       0000001c  0000000000400200  0000000000400200  00000200  2**0
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  1 .note.ABI-tag 00000020  000000000040021c  000000000040021c  0000021c  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  2 .hash         00000028  0000000000400240  0000000000400240  00000240  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  3 .dynsym       00000078  0000000000400268  0000000000400268  00000268  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  4 .dynstr       00000047  00000000004002e0  00000000004002e0  000002e0  2**0
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  5 .gnu.version  0000000a  0000000000400328  0000000000400328  00000328  2**1
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  6 .gnu.version_r 00000020  0000000000400338  0000000000400338  00000338  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  7 .rela.dyn     00000030  0000000000400358  0000000000400358  00000358  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  8 .rela.plt     00000030  0000000000400388  0000000000400388  00000388  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
  9 .init         0000001a  00000000004003b8  00000000004003b8  000003b8  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
 10 .plt          00000030  00000000004003e0  00000000004003e0  000003e0  2**4
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
 11 .plt.got      00000008  0000000000400410  0000000000400410  00000410  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
 12 .text         00000182  0000000000400420  0000000000400420  00000420  2**4
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
 13 .fini         00000009  00000000004005a4  00000000004005a4  000005a4  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, CODE
 14 .rodata       0000000f  00000000004005b0  00000000004005b0  000005b0  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
 15 .eh_frame_hdr 00000034  00000000004005c0  00000000004005c0  000005c0  2**2
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
 16 .eh_frame     000000f4  00000000004005f8  00000000004005f8  000005f8  2**3
                  CONTENTS, ALLOC, LOAD, READONLY, DATA
 17 .init_array   00000008  00000000006006f0  00000000006006f0  000006f0  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 18 .fini_array   00000008  00000000006006f8  00000000006006f8  000006f8  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 19 .dynamic      000001d0  0000000000600700  0000000000600700  00000700  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 20 .got          00000008  00000000006008d0  00000000006008d0  000008d0  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 21 .got.plt      00000028  00000000006008d8  00000000006008d8  000008d8  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 22 .data         00000010  0000000000600900  0000000000600900  00000900  2**3
                  CONTENTS, ALLOC, LOAD, DATA
 23 .bss          00001020  0000000000600920  0000000000600920  00000910  2**5
                  ALLOC
 24 .comment      00000029  0000000000000000  0000000000000000  00000910  2**0
                  CONTENTS, READONLY
 25 .debug_aranges 00000030  0000000000000000  0000000000000000  00000939  2**0
                  CONTENTS, READONLY, DEBUGGING
 26 .debug_info   00000351  0000000000000000  0000000000000000  00000969  2**0
                  CONTENTS, READONLY, DEBUGGING
 27 .debug_abbrev 000000fa  0000000000000000  0000000000000000  00000cba  2**0
                  CONTENTS, READONLY, DEBUGGING
 28 .debug_line   000000ee  0000000000000000  0000000000000000  00000db4  2**0
                  CONTENTS, READONLY, DEBUGGING
 29 .debug_str    000002b8  0000000000000000  0000000000000000  00000ea2  2**0
                  CONTENTS, READONLY, DEBUGGING
SYMBOL TABLE:
0000000000400200 l    d  .interp    0000000000000000              .interp
000000000040021c l    d  .note.ABI-tag  0000000000000000              .note.ABI-tag
0000000000400240 l    d  .hash  0000000000000000              .hash
0000000000400268 l    d  .dynsym    0000000000000000              .dynsym
00000000004002e0 l    d  .dynstr    0000000000000000              .dynstr
0000000000400328 l    d  .gnu.version   0000000000000000              .gnu.version
0000000000400338 l    d  .gnu.version_r 0000000000000000              .gnu.version_r
0000000000400358 l    d  .rela.dyn  0000000000000000              .rela.dyn
0000000000400388 l    d  .rela.plt  0000000000000000              .rela.plt
00000000004003b8 l    d  .init  0000000000000000              .init
00000000004003e0 l    d  .plt   0000000000000000              .plt
0000000000400410 l    d  .plt.got   0000000000000000              .plt.got
0000000000400420 l    d  .text  0000000000000000              .text
00000000004005a4 l    d  .fini  0000000000000000              .fini
00000000004005b0 l    d  .rodata    0000000000000000              .rodata
00000000004005c0 l    d  .eh_frame_hdr  0000000000000000              .eh_frame_hdr
00000000004005f8 l    d  .eh_frame  0000000000000000              .eh_frame
00000000006006f0 l    d  .init_array    0000000000000000              .init_array
00000000006006f8 l    d  .fini_array    0000000000000000              .fini_array
0000000000600700 l    d  .dynamic   0000000000000000              .dynamic
00000000006008d0 l    d  .got   0000000000000000              .got
00000000006008d8 l    d  .got.plt   0000000000000000              .got.plt
0000000000600900 l    d  .data  0000000000000000              .data
0000000000600920 l    d  .bss   0000000000000000              .bss
0000000000000000 l    d  .comment   0000000000000000              .comment
0000000000000000 l    d  .debug_aranges 0000000000000000              .debug_aranges
0000000000000000 l    d  .debug_info    0000000000000000              .debug_info
0000000000000000 l    d  .debug_abbrev  0000000000000000              .debug_abbrev
0000000000000000 l    d  .debug_line    0000000000000000              .debug_line
0000000000000000 l    d  .debug_str 0000000000000000              .debug_str
0000000000000000 l    df *ABS*  0000000000000000              crtstuff.c
0000000000400450 l     F .text  0000000000000000              deregister_tm_clones
0000000000400480 l     F .text  0000000000000000              register_tm_clones
00000000004004c0 l     F .text  0000000000000000              __do_global_dtors_aux
0000000000600928 l     O .bss   0000000000000001              completed.6981
00000000006006f8 l     O .fini_array    0000000000000000              __do_global_dtors_aux_fini_array_entry
00000000004004f0 l     F .text  0000000000000000              frame_dummy
00000000006006f0 l     O .init_array    0000000000000000              __frame_dummy_init_array_entry
0000000000000000 l    df *ABS*  0000000000000000              segments_2.cc
0000000000000000 l    df *ABS*  0000000000000000              crtstuff.c
```

```
00000000004006e8 l     O .eh_frame  0000000000000000              __FRAME_END__
0000000000000000 l    df *ABS*  0000000000000000
00000000006006f8 l       .init_array    0000000000000000              __init_array_end
0000000000600700 l     O .dynamic   0000000000000000              _DYNAMIC
00000000006006f0 l       .init_array    0000000000000000              __init_array_start
00000000004005c0 l       .eh_frame_hdr  0000000000000000              __GNU_EH_FRAME_HDR
00000000006008d8 l     O .got.plt   0000000000000000              _GLOBAL_OFFSET_TABLE_
00000000004005a0 g     F .text  0000000000000002              __libc_csu_fini
0000000000000000  w      *UND*  0000000000000000              _ITM_deregisterTMCloneTable
0000000000600920 g     O .bss   0000000000000008              stdout@@GLIBC_2.2.5
0000000000600900  w      .data  0000000000000000              data_start
0000000000600910 g       .data  0000000000000000              _edata
00000000004005a4 g     F .fini  0000000000000000              _fini
0000000000000000       F *UND*  0000000000000000              __libc_start_main@@GLIBC_2.2.5
0000000000600900 g       .data  0000000000000000              __data_start
0000000000000000       F *UND*  0000000000000000              fprintf@@GLIBC_2.2.5
0000000000000000  w      *UND*  0000000000000000              __gmon_start__
0000000000600940 g     O .bss   0000000000001000              value
0000000000600908 g     O .data  0000000000000000              .hidden __dso_handle
00000000004005b0 g     O .rodata    0000000000000004              _IO_stdin_used
0000000000400530 g     F .text  0000000000000065              __libc_csu_init
0000000000601940 g       .bss   0000000000000000              _end
0000000000400420 g     F .text  000000000000002a              _start
0000000000600910 g       .bss   0000000000000000              __bss_start
00000000004004f7 g     F .text  000000000000002a              main
0000000000600910 g     O .data  0000000000000000              .hidden __TMC_END__
0000000000000000  w      *UND*  0000000000000000              _ITM_registerTMCloneTable
00000000004003b8 g     F .init  0000000000000000              _init

```

implementation difference (side-by-side diff, with suppression)

```
./bin/segments_1                               | ./bin/segments_2
start address 0x0000000000400470                       | start address 0x0000000000400420
         filesz 0x000000000000074c memsz 0x000000000000074c flags r-x    |          filesz 0x00000000000006ec memsz 0x00000000000006ec flags r-x
    LOAD off    0x0000000000000750 vaddr 0x0000000000600750 paddr 0x00   |     LOAD off    0x00000000000006f0 vaddr 0x00000000006006f0 paddr 0x00
         filesz 0x0000000000000228 memsz 0x0000000000001250 flags rw-    |          filesz 0x0000000000000220 memsz 0x0000000000001250 flags rw-
 DYNAMIC off    0x0000000000000760 vaddr 0x0000000000600760 paddr 0x00   |  DYNAMIC off    0x0000000000000700 vaddr 0x0000000000600700 paddr 0x00
EH_FRAME off    0x0000000000000620 vaddr 0x0000000000400620 paddr 0x00   | EH_FRAME off    0x00000000000005c0 vaddr 0x00000000004005c0 paddr 0x00
  INIT                 0x00000000004003f8                  |   INIT                 0x00000000004003b8
  FINI                 0x0000000000400604                  |   FINI                 0x00000000004005a4
  INIT_ARRAY           0x0000000000600750                  |   INIT_ARRAY           0x00000000006006f0
  FINI_ARRAY           0x0000000000600758                  |   FINI_ARRAY           0x00000000006006f8
  STRTAB               0x0000000000400300                  |   STRTAB               0x00000000004002e0
  SYMTAB               0x0000000000400270                  |   SYMTAB               0x0000000000400268
  STRSZ                0x000000000000004e                  |   STRSZ                0x0000000000000047
  PLTGOT               0x0000000000600938                  |   PLTGOT               0x00000000006008d8
  PLTRELSZ             0x0000000000000048                  |   PLTRELSZ             0x0000000000000030
  JMPREL               0x00000000004003b0                  |   JMPREL               0x0000000000400388
  RELA                 0x0000000000400380                  |   RELA                 0x0000000000400358
  VERNEED              0x0000000000400360                  |   VERNEED              0x0000000000400338
  VERSYM               0x000000000040034e                  |   VERSYM               0x0000000000400328
  2 .hash         0000002c  0000000000400240  0000000000400240  000002   |   2 .hash         00000028  0000000000400240  0000000000400240  000002
  3 .dynsym       00000090  0000000000400270  0000000000400270  000002   |   3 .dynsym       00000078  0000000000400268  0000000000400268  000002
  4 .dynstr       0000004e  0000000000400300  0000000000400300  000003   |   4 .dynstr       00000047  00000000004002e0  00000000004002e0  000002
  5 .gnu.version  0000000c  000000000040034e  000000000040034e  000003   |   5 .gnu.version  0000000a  0000000000400328  0000000000400328  000003
  6 .gnu.version_r 00000020  0000000000400360  0000000000400360  00000   |   6 .gnu.version_r 00000020  0000000000400338  0000000000400338  00000
  7 .rela.dyn     00000030  0000000000400380  0000000000400380  000003   |   7 .rela.dyn     00000030  0000000000400358  0000000000400358  000003
  8 .rela.plt     00000048  00000000004003b0  00000000004003b0  000003   |   8 .rela.plt     00000030  0000000000400388  0000000000400388  000003
  9 .init         0000001a  00000000004003f8  00000000004003f8  000003   |   9 .init         0000001a  00000000004003b8  00000000004003b8  000003
 10 .plt          00000040  0000000000400420  0000000000400420  000004   |  10 .plt          00000030  00000000004003e0  00000000004003e0  000003
 11 .plt.got      00000008  0000000000400460  0000000000400460  000004   |  11 .plt.got      00000008  0000000000400410  0000000000400410  000004
 12 .text         00000192  0000000000400470  0000000000400470  000004   |  12 .text         00000182  0000000000400420  0000000000400420  000004
 13 .fini         00000009  0000000000400604  0000000000400604  000006   |  13 .fini         00000009  00000000004005a4  00000000004005a4  000005
 14 .rodata       0000000f  0000000000400610  0000000000400610  000006   |  14 .rodata       0000000f  00000000004005b0  00000000004005b0  000005
 15 .eh_frame_hdr 00000034  0000000000400620  0000000000400620  000006   |  15 .eh_frame_hdr 00000034  00000000004005c0  00000000004005c0  000005
 16 .eh_frame     000000f4  0000000000400658  0000000000400658  000006   |  16 .eh_frame     000000f4  00000000004005f8  00000000004005f8  000005
 17 .init_array   00000008  0000000000600750  0000000000600750  000007   |  17 .init_array   00000008  00000000006006f0  00000000006006f0  000006
 18 .fini_array   00000008  0000000000600758  0000000000600758  000007   |  18 .fini_array   00000008  00000000006006f8  00000000006006f8  000006
 19 .dynamic      000001d0  0000000000600760  0000000000600760  000007   |  19 .dynamic      000001d0  0000000000600700  0000000000600700  000007
 20 .got          00000008  0000000000600930  0000000000600930  000009   |  20 .got          00000008  00000000006008d0  00000000006008d0  000008
 21 .got.plt      00000030  0000000000600938  0000000000600938  000009   |  21 .got.plt      00000028  00000000006008d8  00000000006008d8  000008
 22 .data         00000010  0000000000600968  0000000000600968  000009   |  22 .data         00000010  0000000000600900  0000000000600900  000009
 23 .bss          00001020  0000000000600980  0000000000600980  000009   |  23 .bss          00001020  0000000000600920  0000000000600920  000009
 24 .comment      00000029  0000000000000000  0000000000000000  000009   |  24 .comment      00000029  0000000000000000  0000000000000000  000009
 25 .debug_aranges 00000030  0000000000000000  0000000000000000  00000   |  25 .debug_aranges 00000030  0000000000000000  0000000000000000  00000
 26 .debug_info   00000351  0000000000000000  0000000000000000  000009   |  26 .debug_info   00000351  0000000000000000  0000000000000000  000009
 27 .debug_abbrev 000000fa  0000000000000000  0000000000000000  00000d   |  27 .debug_abbrev 000000fa  0000000000000000  0000000000000000  00000c
 28 .debug_line   000000f0  0000000000000000  0000000000000000  00000e   |  28 .debug_line   000000ee  0000000000000000  0000000000000000  00000d
 29 .debug_str    000002b8  0000000000000000  0000000000000000  00000f   |  29 .debug_str    000002b8  0000000000000000  0000000000000000  00000e
0000000000400270 l    d  .dynsym    0000000000000000                  | 0000000000400268 l    d  .dynsym    0000000000000000
0000000000400300 l    d  .dynstr    0000000000000000                  | 00000000004002e0 l    d  .dynstr    0000000000000000
000000000040034e l    d  .gnu.version   0000000000000000                  | 0000000000400328 l    d  .gnu.version   0000000000000000
0000000000400360 l    d  .gnu.version_r 0000000000000000                  | 0000000000400338 l    d  .gnu.version_r 0000000000000000
0000000000400380 l    d  .rela.dyn  0000000000000000                  | 0000000000400358 l    d  .rela.dyn  0000000000000000
00000000004003b0 l    d  .rela.plt  0000000000000000                  | 0000000000400388 l    d  .rela.plt  0000000000000000
00000000004003f8 l    d  .init  0000000000000000              .init   | 00000000004003b8 l    d  .init  0000000000000000              .init
0000000000400420 l    d  .plt   0000000000000000              .plt    | 00000000004003e0 l    d  .plt   0000000000000000              .plt
0000000000400460 l    d  .plt.got   0000000000000000                  | 0000000000400410 l    d  .plt.got   0000000000000000
0000000000400470 l    d  .text  0000000000000000              .text   | 0000000000400420 l    d  .text  0000000000000000              .text
0000000000400604 l    d  .fini  0000000000000000              .fini   | 00000000004005a4 l    d  .fini  0000000000000000              .fini
0000000000400610 l    d  .rodata    0000000000000000                  | 00000000004005b0 l    d  .rodata    0000000000000000
0000000000400620 l    d  .eh_frame_hdr  0000000000000000                  | 00000000004005c0 l    d  .eh_frame_hdr  0000000000000000
0000000000400658 l    d  .eh_frame  0000000000000000                  | 00000000004005f8 l    d  .eh_frame  0000000000000000
0000000000600750 l    d  .init_array    0000000000000000                  | 00000000006006f0 l    d  .init_array    0000000000000000
0000000000600758 l    d  .fini_array    0000000000000000                  | 00000000006006f8 l    d  .fini_array    0000000000000000
0000000000600760 l    d  .dynamic   0000000000000000                  | 0000000000600700 l    d  .dynamic   0000000000000000
0000000000600930 l    d  .got   0000000000000000              .got    | 00000000006008d0 l    d  .got   0000000000000000              .got
0000000000600938 l    d  .got.plt   0000000000000000                  | 00000000006008d8 l    d  .got.plt   0000000000000000
0000000000600968 l    d  .data  0000000000000000              .data   | 0000000000600900 l    d  .data  0000000000000000              .data
0000000000600980 l    d  .bss   0000000000000000              .bss    | 0000000000600920 l    d  .bss   0000000000000000              .bss
00000000004004a0 l     F .text  0000000000000000              deregist    | 0000000000400450 l     F .text  0000000000000000              deregist
00000000004004d0 l     F .text  0000000000000000              register    | 0000000000400480 l     F .text  0000000000000000              register
0000000000400510 l     F .text  0000000000000000              __do_glo    | 00000000004004c0 l     F .text  0000000000000000              __do_glo
0000000000600988 l     O .bss   0000000000000001              complete    | 0000000000600928 l     O .bss   0000000000000001              complete
0000000000600758 l     O .fini_array    0000000000000000                  | 00000000006006f8 l     O .fini_array    0000000000000000
0000000000400540 l     F .text  0000000000000000              frame_du    | 00000000004004f0 l     F .text  0000000000000000              frame_du
0000000000600750 l     O .init_array    0000000000000000                  | 00000000006006f0 l     O .init_array    0000000000000000
0000000000000000 l    df *ABS*  0000000000000000              segments    | 0000000000000000 l    df *ABS*  0000000000000000              segments
0000000000400748 l     O .eh_frame  0000000000000000                  | 00000000004006e8 l     O .eh_frame  0000000000000000
0000000000600758 l       .init_array    0000000000000000                  | 00000000006006f8 l       .init_array    0000000000000000
0000000000600760 l     O .dynamic   0000000000000000                  | 0000000000600700 l     O .dynamic   0000000000000000
0000000000600750 l       .init_array    0000000000000000                  | 00000000006006f0 l       .init_array    0000000000000000
0000000000400620 l       .eh_frame_hdr  0000000000000000                  | 00000000004005c0 l       .eh_frame_hdr  0000000000000000
0000000000600938 l     O .got.plt   0000000000000000                  | 00000000006008d8 l     O .got.plt   0000000000000000
0000000000400600 g     F .text  0000000000000002              __libc_c    | 00000000004005a0 g     F .text  0000000000000002              __libc_c
0000000000600980 g     O .bss   0000000000000008              stdout@@    | 0000000000600920 g     O .bss   0000000000000008              stdout@@
0000000000600968  w      .data  0000000000000000              data_sta    | 0000000000600900  w      .data  0000000000000000              data_sta
0000000000600978 g       .data  0000000000000000              _edata      | 0000000000600910 g       .data  0000000000000000              _edata
0000000000400604 g     F .fini  0000000000000000              _fini   | 00000000004005a4 g     F .fini  0000000000000000              _fini
0000000000000000       F *UND*  0000000000000000              memset@@    <
0000000000600968 g       .data  0000000000000000              __data_s    | 0000000000600900 g       .data  0000000000000000              __data_s
00000000006009a0 g     O .bss   0000000000001000              value   | 0000000000600940 g     O .bss   0000000000001000              value
0000000000600970 g     O .data  0000000000000000              .hidden     | 0000000000600908 g     O .data  0000000000000000              .hidden
0000000000400610 g     O .rodata    0000000000000004                  | 00000000004005b0 g     O .rodata    0000000000000004
0000000000400590 g     F .text  0000000000000065              __libc_c    | 0000000000400530 g     F .text  0000000000000065              __libc_c
00000000006019a0 g       .bss   0000000000000000              _end    | 0000000000601940 g       .bss   0000000000000000              _end
0000000000400470 g     F .text  000000000000002a              _start      | 0000000000400420 g     F .text  000000000000002a              _start
0000000000600978 g       .bss   0000000000000000              __bss_st    | 0000000000600910 g       .bss   0000000000000000              __bss_st
0000000000400547 g     F .text  000000000000003e              main    | 00000000004004f7 g     F .text  000000000000002a              main
0000000000600978 g     O .data  0000000000000000              .hidden     | 0000000000600910 g     O .data  0000000000000000              .hidden
00000000004003f8 g     F .init  0000000000000000              _init   | 00000000004003b8 g     F .init  0000000000000000              _init

== No Difference ==
```