Eax – current loop iteration

(rbp – 0x8) -> Max loop

Jl – jump if lesser than max loop

Cdqe – convert double word to quadword

```
   0x40062d <main+102>       cmp    eax, DWORD PTR [rbp-0x8]
→  0x400630 <main+105>       jl     0x40060e <main+71>        TAKEN [Reason: S!=O]
 ↳    0x40060e <main+71>        mov    eax, DWORD PTR [rbp-0x4]
      0x400611 <main+74>        cdqe
      0x400613 <main+76>        movzx  eax, BYTE PTR [rbp+rax*1-0x10]
      0x400618 <main+81>        mov    BYTE PTR [rbp-0x9], al
      0x40061b <main+84>        movsx  eax, BYTE PTR [rbp-0x9]
      0x40061f <main+88>        mov    edi, eax
─────────────────────────────────────────────────────────────────────────────
    8       int max_count = strlen(greeting);
    9
   10       printf("Press enter....");
   11       getchar();                          // Get any character input and pro
   12
            // count=0x0, max_count=0x6
→  13       for(count = 0; count < max_count; count++) {
```

1st Rbp – 0x4 -> current loop iteration

2nd memory to register eax

3rd compare current loop iteration with max counter

4th jump if lesser

```
0x400626 <main+95>        add    DWORD PTR [rbp-0x4], 0x1
0x40062a <main+99>        mov    eax, DWORD PTR [rbp-0x4]
0x40062d <main+102>       cmp    eax, DWORD PTR [rbp-0x8]
0x400630 <main+105>       jl     0x40060e <main+71>
```

If both eax and rbp -0x8 tally, zero flag will be set and jl wouldn't be taken

```
gef➤  print $eflags
$1 = [ PF ZF IF ]
gef➤
```