

```

1  int main() {
2      int counter = 0;
3      for (int i = 0; i < 100; i++) {
4          counter++;
5      }
6  }
7
8
9  |0x5fa <main>      push    %rbp           // save the old base pointer on the stack
10 |0x5fb <main+1>     mov     %rsp,%rbp      // set the new base pointer to the stack pointer
11 |0x5fe <main+4>     movl    $0x0,-0x8(%rbp) // use -0x8(%rbp) as counter var: int counter = 0;
12 |0x605 <main+11>    movl    $0x0,-0x4(%rbp) // use -0x4(%rbp) as i var: int i = 0;
13 |0x60c <main+18>    jmp     0x616 <main+28> // jump to the loop code (instruction 0x616)
14 |0x60e <main+20>    addl    $0x1,-0x8(%rbp) // add 1 to the counter var: counter++;
15 |0x612 <main+24>    addl    $0x1,-0x4(%rbp) // add 1 to the i var: i++;
16 |0x616 <main+28>    cmpl    $0x63,-0x4(%rbp) // cmp $0x63 to i var: i <= 99 (compiler changed i < 100 to this)
17 |0x61a <main+32>    jle     0x60e <main+20> // jump to instruction 0x60e if arg2 of cmp is less than or equal arg1
18 |0x61c <main+34>    mov     $0x0,%eax     // move 0 into %eax (the return value)
19 |0x621 <main+39>    pop     %rbp         // reset the base pointer to the value saved on the stack at the start
20 |0x622 <main+40>    retq                    // return :)

```