From logic bomb – Phase 3 in assembly

```
0x0000000000400f47 <+0>: sub    $0x18,%rsp
0x0000000000400f4b <+4>: lea    0x8(%rsp),%rcx
0x0000000000400f50 <+9>: lea    0xc(%rsp),%rdx
0x0000000000400f55 <+14>: mov    $0x40277d,%esi
0x0000000000400f5a <+19>: mov    $0x0,%eax
0x0000000000400f5f <+24>: callq  0x400c10 <__isoc99_sscanf@plt>
0x0000000000400f64 <+29>: cmp    $0x1,%eax
0x0000000000400f67 <+32>: jg     0x400f6e <phase_3+39>
0x0000000000400f69 <+34>: callq  0x401534 <explode_bomb>
0x0000000000400f6e <+39>: cmpl   $0x7,0xc(%rsp)
0x0000000000400f73 <+44>: ja     0x400fb1 <phase_3+106>
0x0000000000400f75 <+46>: mov    0xc(%rsp),%eax
0x0000000000400f79 <+50>: jmpq   *0x4024e0(,%rax,8)
0x0000000000400f80 <+57>: mov    $0x70,%eax
0x0000000000400f85 <+62>: jmp    0x400fc2 <phase_3+123>
0x0000000000400f87 <+64>: mov    $0x141,%eax
0x0000000000400f8c <+69>: jmp    0x400fc2 <phase_3+123>
0x0000000000400f8e <+71>: mov    $0x3ad,%eax
0x0000000000400f93 <+76>: jmp    0x400fc2 <phase_3+123>
0x0000000000400f95 <+78>: mov    $0xd4,%eax
0x0000000000400f9a <+83>: jmp    0x400fc2 <phase_3+123>
0x0000000000400f9c <+85>: mov    $0x27b,%eax
0x0000000000400fa1 <+90>: jmp    0x400fc2 <phase_3+123>
0x0000000000400fa3 <+92>: mov    $0x3bc,%eax
0x0000000000400fa8 <+97>: jmp    0x400fc2 <phase_3+123>
0x0000000000400faa <+99>: mov    $0x29d,%eax
0x0000000000400faf <+104>: jmp    0x400fc2 <phase_3+123>
0x0000000000400fb1 <+106>: callq  0x401534 <explode_bomb>
0x0000000000400fb6 <+111>: mov    $0x0,%eax
0x0000000000400fbb <+116>: jmp    0x400fc2 <phase_3+123>
0x0000000000400fbd <+118>: mov    $0x311,%eax
0x0000000000400fc2 <+123>: cmp    0x8(%rsp),%eax
0x0000000000400fc6 <+127>: je     0x400fcd <phase_3+134>
0x0000000000400fc8 <+129>: callq  0x401534 <explode_bomb>
0x0000000000400fcd <+134>: add    $0x18,%rsp
0x0000000000400fd1 <+138>: retq
```

From logic bomb – Function 4 in assembly

```
Dump of assembler code for function func4:
   0x0000000000400fd2 <+0>: push   %rbx
   0x0000000000400fd3 <+1>: mov    %edx,%eax
   0x0000000000400fd5 <+3>: sub    %esi,%eax
   0x0000000000400fd7 <+5>: mov    %eax,%ebx
   0x0000000000400fd9 <+7>: shr    $0x1f,%ebx
   0x0000000000400fdc <+10>: add    %ebx,%eax
   0x0000000000400fde <+12>: sar    %eax
   0x0000000000400fe0 <+14>: lea    (%rax,%rsi,1),%ebx
   0x0000000000400fe3 <+17>: cmp    %edi,%ebx
   0x0000000000400fe5 <+19>: jle    0x400ff3 <func4+33>
   0x0000000000400fe7 <+21>: lea    -0x1(%rbx),%edx
   0x0000000000400fea <+24>: callq  0x400fd2 <func4>
   0x0000000000400fef <+29>: add    %ebx,%eax
   0x0000000000400ff1 <+31>: jmp    0x401003 <func4+49>
   0x0000000000400ff3 <+33>: mov    %ebx,%eax
   0x0000000000400ff5 <+35>: cmp    %edi,%ebx
   0x0000000000400ff7 <+37>: jge    0x401003 <func4+49>
   0x0000000000400ff9 <+39>: lea    0x1(%rbx),%esi
   0x0000000000400ffc <+42>: callq  0x400fd2 <func4>
   0x0000000000401001 <+47>: add    %ebx,%eax
   0x0000000000401003 <+49>: pop    %rbx
   0x0000000000401004 <+50>: retq
```