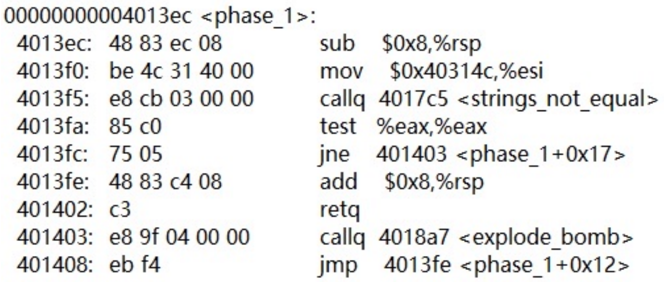
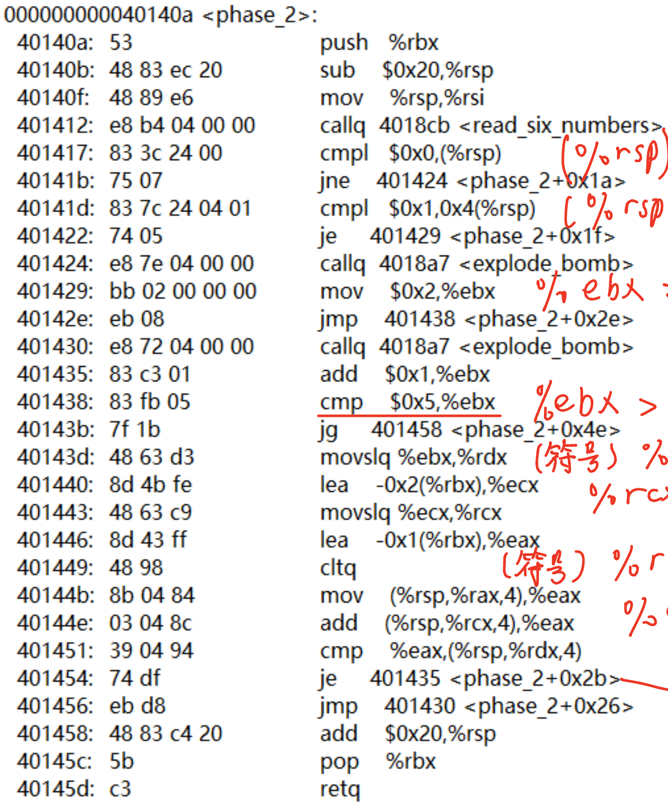


```
00000000004013ec <phase_1>:
  4013ec:  48 83 ec 08          sub    $0x8,%rsp
  4013f0:  be 4c 31 40 00       mov    $0x40314c,%esi
  4013f5:  e8 cb 03 00 00       callq  4017c5 <strings_not_equal>
  4013fa:  85 c0                test   %eax,%eax
  4013fc:  75 05                jne    401403 <phase_1+0x17>
  4013fe:  48 83 c4 08          add    $0x8,%rsp
  401402:  c3                   retq
  401403:  e8 9f 04 00 00       callq  4018a7 <explode_bomb>
  401408:  eb f4                jmp    4013fe <phase_1+0x12>
```

Crikey! I have lost my mojo!

0, 1, 1, 2, 3, 5



```
000000000040140a <phase_2>:
  40140a:  53                   push   %rbx
  40140b:  48 83 ec 20          sub    $0x20,%rsp
  40140f:  48 89 e6             mov    %rsp,%rsi
  401412:  e8 b4 04 00 00       callq  4018cb <read_six_numbers>
  401417:  83 3c 24 00          cmpl   $0x0,(%rsp)          (%rsp)!= 0 , bomb
  40141b:  75 07                jne    401424 <phase_2+0x1a>
  40141d:  83 7c 24 04 01       cmpl   $0x1,0x4(%rsp)       (%rsp+4) != 1 , bomb
  401422:  74 05                je     401429 <phase_2+0x1f>
  401424:  e8 7e 04 00 00       callq  4018a7 <explode_bomb>
  401429:  bb 02 00 00 00       mov    $0x2,%ebx            %ebx = 2
  40142e:  eb 08                jmp    401438 <phase_2+0x2e>
  401430:  e8 72 04 00 00       callq  4018a7 <explode_bomb>     ebx++;
  401435:  83 c3 01             add    $0x1,%ebx
  401438:  83 fb 05             cmp    $0x5,%ebx            %ebx > 5, 成功.
  40143b:  7f 1b                jg     401458 <phase_2+0x4e>
  40143d:  48 63 d3             movslq %ebx,%rdx            (符号) %rdx = %ebx
  401440:  8d 4b fe             lea    -0x2(%rbx),%ecx        %rcx = %rbx -2
  401443:  48 63 c9             movslq %ecx,%rcx
  401446:  8d 43 ff             lea    -0x1(%rbx),%eax
  401449:  48 98                cltq                    (符号) %rax = %eax = %rbx -1
  40144b:  8b 04 84             mov    (%rsp,%rax,4),%eax
  40144e:  03 04 8c             add    (%rsp,%rcx,4),%eax     %eax = (%rsp + 4rax) + (rsp+4rcx)
  401451:  39 04 94             cmp    %eax,(%rsp,%rdx,4)
  401454:  74 df                je     401435 <phase_2+0x2b> ———— ebx == 3
  401456:  eb d8                jmp    401430 <phase_2+0x26>
  401458:  48 83 c4 20          add    $0x20,%rsp
  40145c:  5b                   pop    %rbx
  40145d:  c3                   retq
```

```
000000000040145e <phase_3>:
  40145e:  48 83 ec 18             sub    $0x18,%rsp
  401462:  48 8d 4c 24 08          lea    0x8(%rsp),%rcx
  401467:  48 8d 54 24 0c          lea    0xc(%rsp),%rdx
  40146c:  be 37 33 40 00          mov    $0x403337,%esi
  401471:  b8 00 00 00 00          mov    $0x0,%eax
  401476:  e8 95 fc ff ff          callq  401110 <__isoc99_sscanf@plt>
  40147b:  83 f8 01                cmp    $0x1,%eax
  40147e:  7e 12                   jle    401492 <phase_3+0x34>
  401480:  8b 44 24 0c             mov    0xc(%rsp),%eax
  401484:  83 f8 07                cmp    $0x7,%eax
  401487:  77 4a                   ja     4014d3 <phase_3+0x75>
  401489:  89 c0                   mov    %eax,%eax
  40148b:  ff 24 c5 80 31 40 00    jmpq   *0x403180(,%rax,8)
  401492:  e8 10 04 00 00          callq  4018a7 <explode_bomb>
  401497:  eb e7                   jmp    401480 <phase_3+0x22>
  401499:  b8 c9 03 00 00          mov    $0x3c9,%eax
  40149e:  39 44 24 08             cmp    %eax,0x8(%rsp)
  4014a2:  75 42                   jne    4014e6 <phase_3+0x88>
  4014a4:  48 83 c4 18             add    $0x18,%rsp
  4014a8:  c3                      retq
  4014a9:  b8 75 01 00 00          mov    $0x175,%eax
  4014ae:  eb ee                   jmp    40149e <phase_3+0x40>
  4014b0:  b8 86 02 00 00          mov    $0x286,%eax
  4014b5:  eb e7                   jmp    40149e <phase_3+0x40>
  4014b7:  b8 2b 01 00 00          mov    $0x12b,%eax
  4014bc:  eb e0                   jmp    40149e <phase_3+0x40>
  4014be:  b8 bb 03 00 00          mov    $0x3bb,%eax
  4014c3:  eb d9                   jmp    40149e <phase_3+0x40>
  4014c5:  b8 71 02 00 00          mov    $0x271,%eax
  4014ca:  eb d2                   jmp    40149e <phase_3+0x40>
  4014cc:  b8 6d 03 00 00          mov    $0x36d,%eax
  4014d1:  eb cb                   jmp    40149e <phase_3+0x40>
  4014d3:  e8 cf 03 00 00          callq  4018a7 <explode_bomb>
  4014d8:  b8 00 00 00 00          mov    $0x0,%eax
  4014dd:  eb bf                   jmp    40149e <phase_3+0x40>
  4014df:  b8 f7 02 00 00          mov    $0x2f7,%eax
  4014e4:  eb b8                   jmp    40149e <phase_3+0x40>
  4014e6:  e8 bc 03 00 00          callq  4018a7 <explode_bomb>
  4014eb:  eb b7                   jmp    4014a4 <phase_3+0x46>
```

2, 3, 4, 5, 6, 7

1<   <=7

eax <= 1 , bomb

eax = rsp+12

eax > 7 , bomb

%eax != (%rsp + 8) , bomb

%eax = (%rsp + 8)、返回

2	373
3	646
4	299
5	955
6	625
7	877

8 35

```
000000000040151f <phase_4>:
  40151f:  48 83 ec 18           sub    $0x18,%rsp
  401523:  48 8d 4c 24 08        lea    0x8(%rsp),%rcx
  401528:  48 8d 54 24 0c        lea    0xc(%rsp),%rdx
  40152d:  be 37 33 40 00        mov    $0x403337,%esi          → %d %d
  401532:  b8 00 00 00 00        mov    $0x0,%eax               eax = 0
  401537:  e8 d4 fb ff ff        callq  401110 <__isoc99_sscanf@plt>        scanf.
  40153c:  83 f8 02              cmp    $0x2,%eax               eax != 2, bomb
  40153f:  75 0d                 jne    40154e <phase_4+0x2f>
  401541:  8b 44 24 0c           mov    0xc(%rsp),%eax          eax = (rsp + 12)
  401545:  85 c0                 test   %eax,%eax
  401547:  78 05                 js     40154e <phase_4+0x2f>   eax < 0 , bomb
  401549:  83 f8 0e              cmp    $0xe,%eax               0<= eax <= 14
  40154c:  7e 05                 jle    401553 <phase_4+0x34>
  40154e:  e8 54 03 00 00        callq  4018a7 <explode_bomb>   eax > 14, bomb
  401553:  ba 0e 00 00 00        mov    $0xe,%edx
  401558:  be 00 00 00 00        mov    $0x0,%esi               14 , 0 , edi
  40155d:  8b 7c 24 0c           mov    0xc(%rsp),%edi          第1个输入        2入：35
  401561:  e8 87 ff ff ff        callq  4014ed <func4>
  401566:  83 f8 23              cmp    $0x23,%eax
  401569:  75 07                 jne    401572 <phase_4+0x53>   eax != 35, bomb
  40156b:  83 7c 24 08 23        cmpl   $0x23,0x8(%rsp)
  401570:  74 05                 je     401577 <phase_4+0x58>   (rsp + 8) == 35 , 成功
  401572:  e8 30 03 00 00        callq  4018a7 <explode_bomb>
  401577:  48 83 c4 18           add    $0x18,%rsp
  40157b:  c3                    retq
```

edx/x esi/y edi   设 eax 为 m , ebx 为 n

```
00000000004014ed <func4>:
  4014ed:  53                    push   %rbx
  4014ee:  89 d0        m = x-y  mov    %edx,%eax          eax = edx - esi = 14
  4014f0:  29 f0                 sub    %esi,%eax
  4014f2:  89 c3        n = m    mov    %eax,%ebx          ebx = eax >> 31 + eax
  4014f4:  c1 eb 1f              shr    $0x1f,%ebx                (逻辑)
  4014f7:  01 c3    n = n >>31 + m  add  %eax,%ebx
  4014f9:  d1 fb        逻辑     sar    %ebx               ebx = ebx >> 1     算术
  4014fb:  01 f3        n >>= 1  add    %esi,%ebx          ebx = ebx + esi
  4014fd:  39 fb        算术     cmp    %edi,%ebx
  4014ff:  7f 06                 jg     401507 <func4+0x1a>
  401501:  7c 10        n += y   jl     401513 <func4+0x26>
  401503:  89 d8                 mov    %ebx,%eax             ==
  401505:  5b                    pop    %rbx                             m     t
  401506:  c3       edi / n      retq
→ 401507:  8d 53 ff              lea    -0x1(%rbx),%edx    edx = rbx - 1      ebx > edi
  40150a:  e8 de ff ff ff        callq  4014ed <func4>
  40150f:  01 c3                 add    %eax,%ebx          ebx = ebx + eax
  401511:  eb f0                 jmp    401503 <func4+0x16>
→ 401513:  8d 73 01              lea    0x1(%rbx),%esi     esi = rbx + 1      ebx < edi
  401516:  e8 d2 ff ff ff        callq  4014ed <func4>
  40151b:  01 c3                 add    %eax,%ebx          ebx = ebx + eax
  40151d:  eb e4                 jmp    401503 <func4+0x16>
```

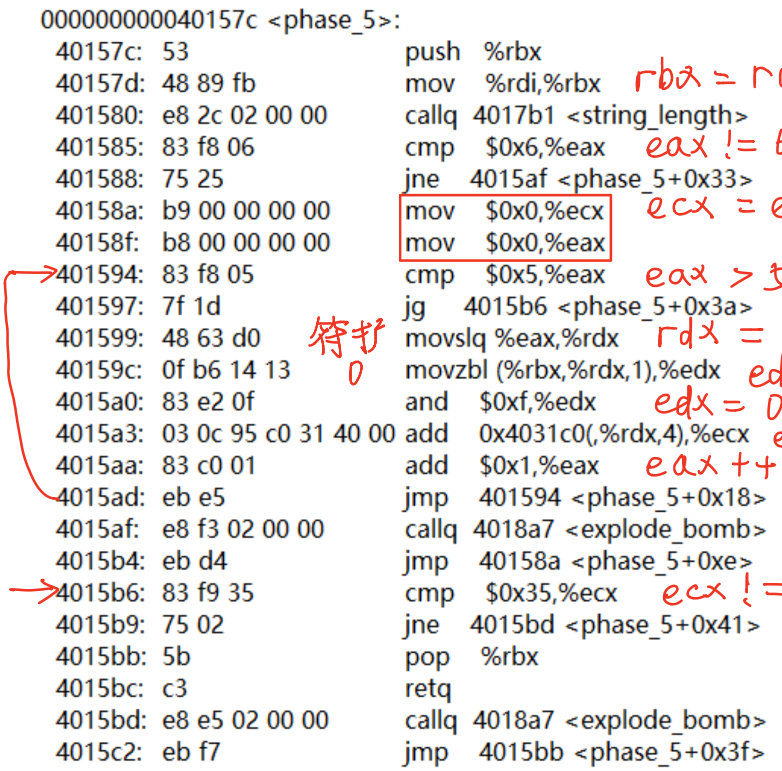
```
#include <stdio.h>

// x in %edx, y in %esi, t in %edi
int func4 (int x, int y, int t) {
    int n = x - y;
    int m = n / 2 + y;
    if (m == t)
        return m;
    else if (m > t)
        return (m + func4 (m - 1, y, t));
    else
        return (m + func4 (x, m + 1, t));
}

int main () {
    int ret, t;
    for (t = 0; t <= 14; t++)
    {
        ret = func4 (14, 0, t);
        printf ("%d ", t);
        if (ret == 35)
            printf("%d\n", t);
        else
            printf("%d wrong\n", ret);
    }
    return 0;
}
```

8/35

string_length : 6



```
000000000040157c <phase_5>:
  40157c:  53                    push   %rbx
  40157d:  48 89 fb              mov    %rdi,%rbx              rbx = rdi
  401580:  e8 2c 02 00 00        callq  4017b1 <string_length>
  401585:  83 f8 06              cmp    $0x6,%eax              eax != 6 , bomb
  401588:  75 25                 jne    4015af <phase_5+0x33>
  40158a:  b9 00 00 00 00        mov    $0x0,%ecx              ecx = eax = 0 .
  40158f:  b8 00 00 00 00        mov    $0x0,%eax
  401594:  83 f8 05              cmp    $0x5,%eax              eax > 5 , jump
  401597:  7f 1d                 jg     4015b6 <phase_5+0x3a>
  401599:  48 63 d0        符扩   movslq %eax,%rdx              rdx = eax
  40159c:  0f b6 14 13      0     movzbl (%rbx,%rdx,1),%edx     edx = (rbx + rdx)
  4015a0:  83 e2 0f              and    $0xf,%edx              edx = 0xf & edx
  4015a3:  03 0c 95 c0 31 40 00  add    0x4031c0(,%rdx,4),%ecx  ecx = array [rdx]. + ecx
  4015aa:  83 c0 01              add    $0x1,%eax              eax ++;
  4015ad:  eb e5                 jmp    401594 <phase_5+0x18>
  4015af:  e8 f3 02 00 00        callq  4018a7 <explode_bomb>
  4015b4:  eb d4                 jmp    40158a <phase_5+0xe>
  4015b6:  83 f9 35              cmp    $0x35,%ecx             ecx != 53 , bomb
  4015b9:  75 02                 jne    4015bd <phase_5+0x41>
  4015bb:  5b                    pop    %rbx
  4015bc:  c3                    retq
  4015bd:  e8 e5 02 00 00        callq  4018a7 <explode_bomb>
  4015c2:  eb f7                 jmp    4015bb <phase_5+0x3f>
```

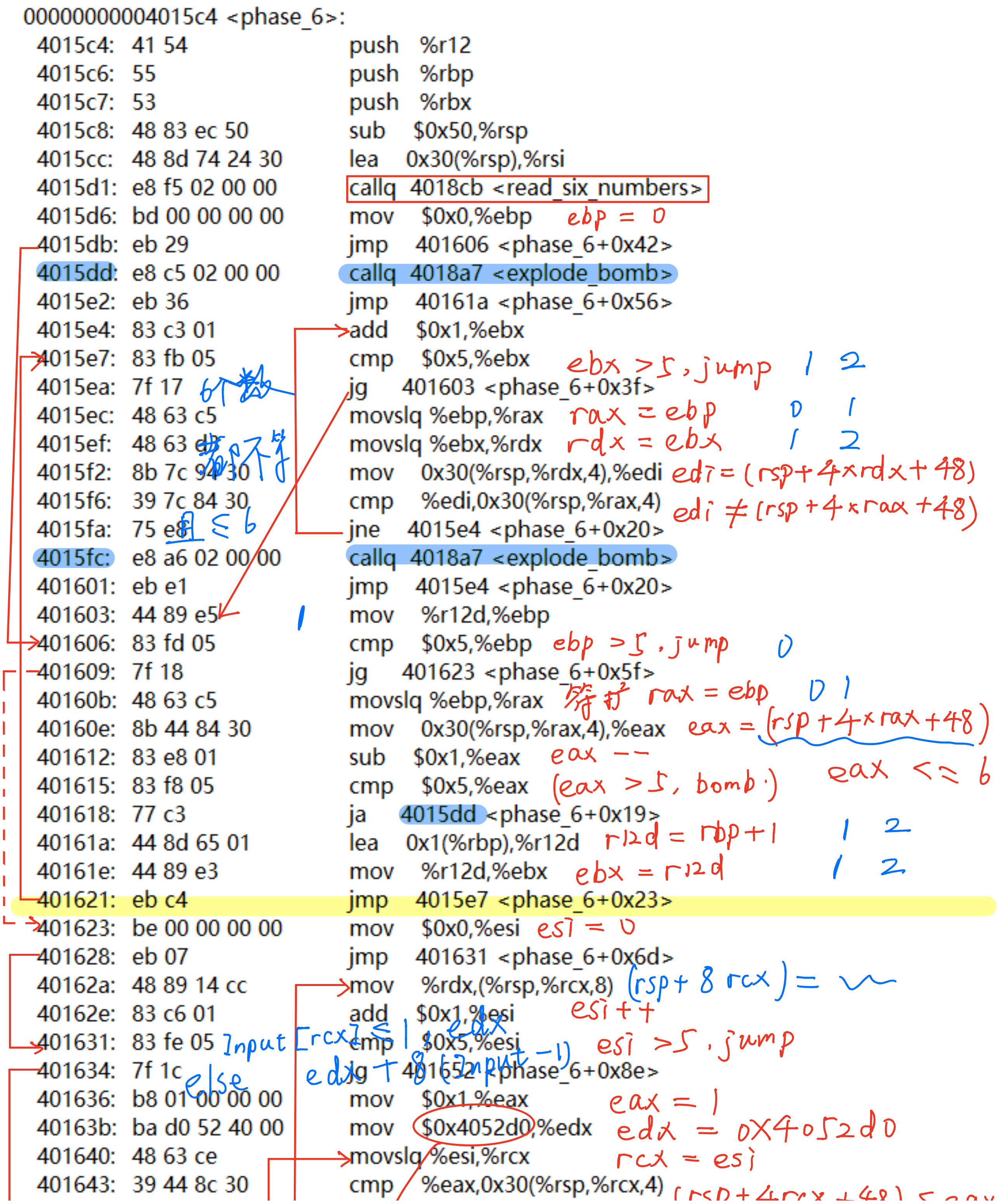
0	1	2	3	4	5	6	7
2,	10,	6,	1,	12,	16,	9,	3
8	9	A	B	C	D	E	F
4,	7,	14,	5,	11,	8,	15,	13

array: 0—15

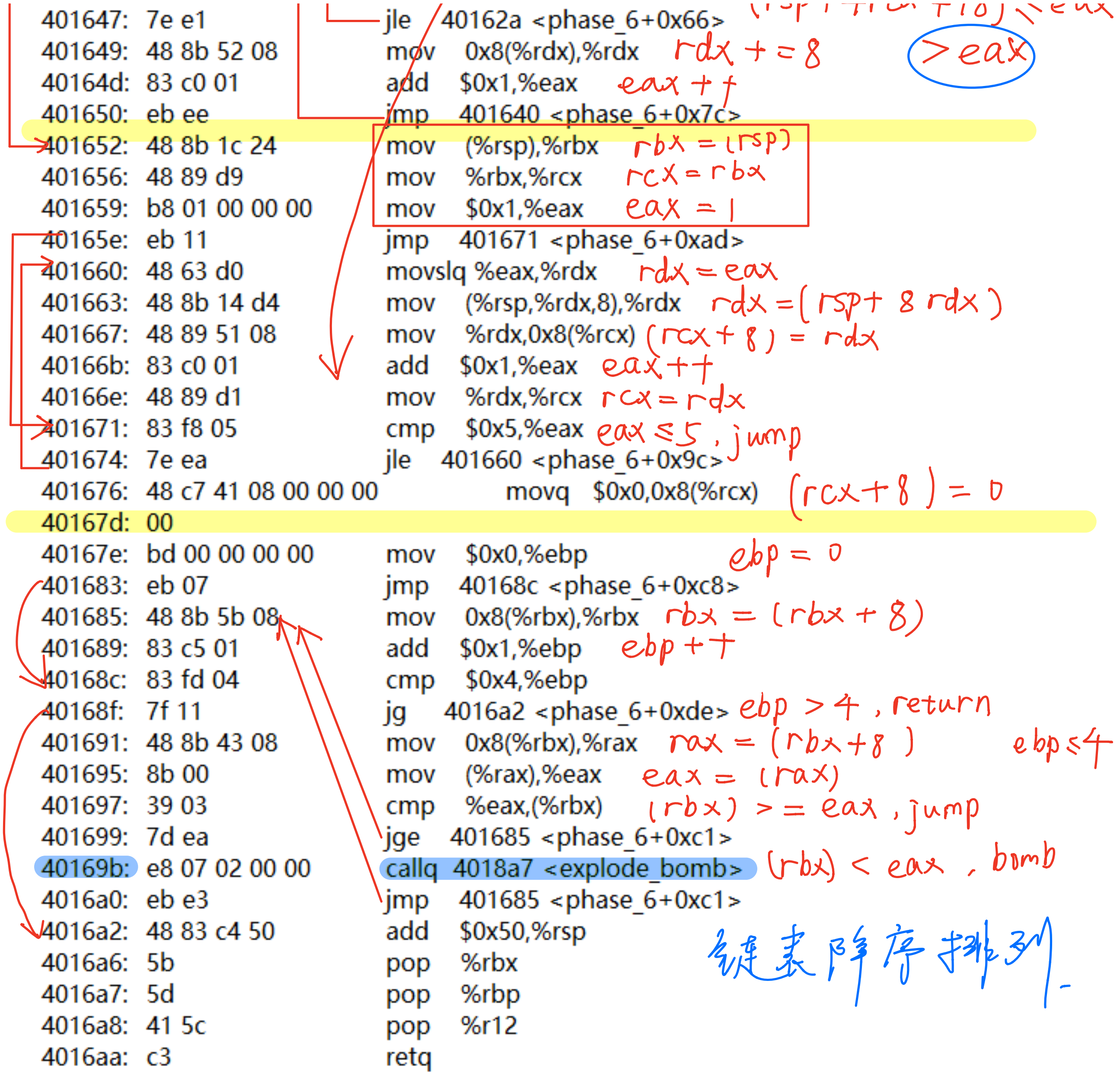
f f f f f m

53   6个数之和

ASCII码二进制后4位



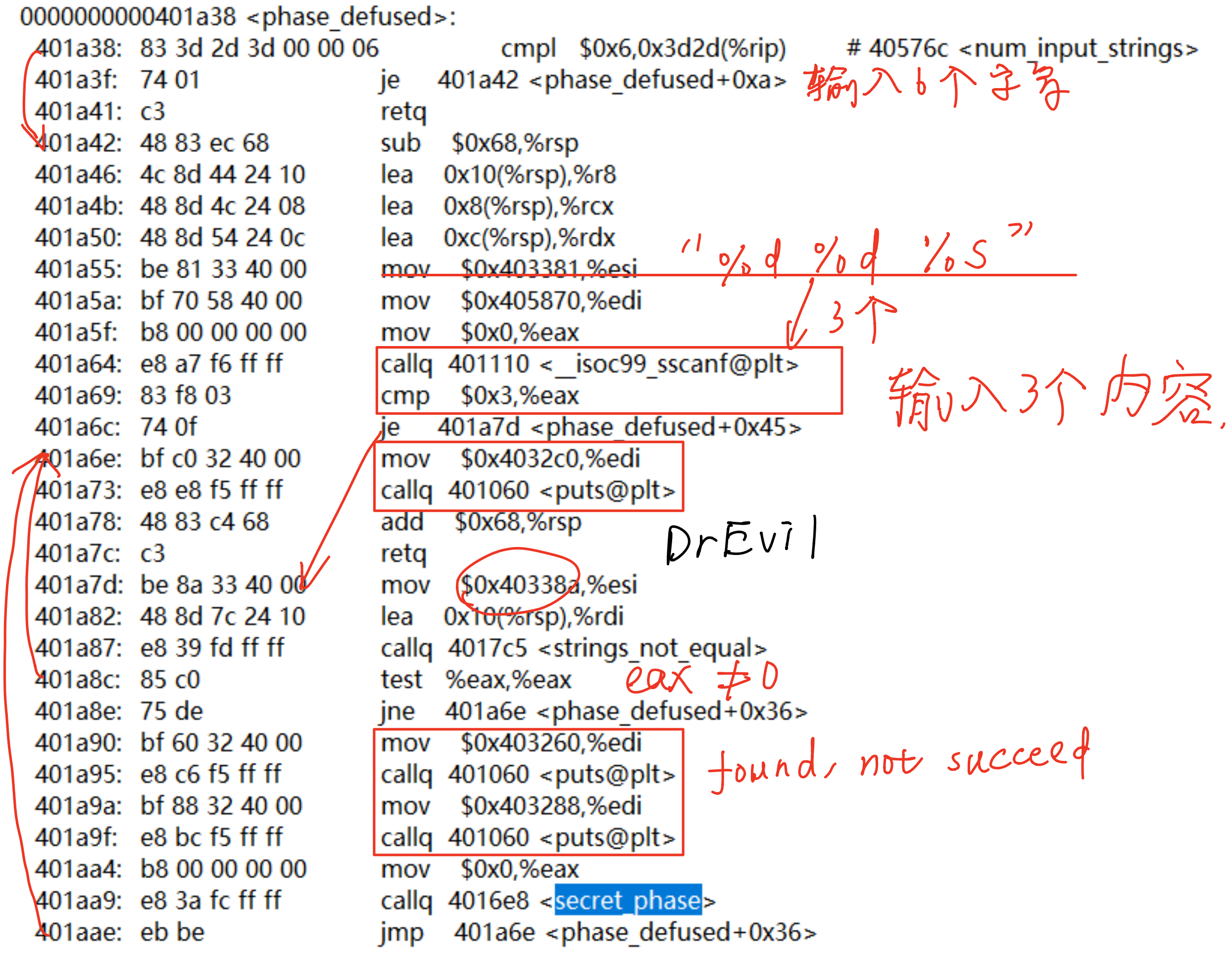
```
00000000004015c4 <phase_6>:
  4015c4:  41 54                 push   %r12
  4015c6:  55                    push   %rbp
  4015c7:  53                    push   %rbx
  4015c8:  48 83 ec 50           sub    $0x50,%rsp
  4015cc:  48 8d 74 24 30        lea    0x30(%rsp),%rsi
  4015d1:  e8 f5 02 00 00        callq  4018cb <read_six_numbers>
  4015d6:  bd 00 00 00 00        mov    $0x0,%ebp              ebp = 0
  4015db:  eb 29                 jmp    401606 <phase_6+0x42>
  4015dd:  e8 c5 02 00 00        callq  4018a7 <explode_bomb>
  4015e2:  eb 36                 jmp    40161a <phase_6+0x56>
  4015e4:  83 c3 01              add    $0x1,%ebx
  4015e7:  83 fb 05              cmp    $0x5,%ebx              ebx >5, jump     1   2
  4015ea:  7f 17     6个数       jg     401603 <phase_6+0x3f>
  4015ec:  48 63 c5              movslq %ebp,%rax              rax = ebp        0   1
  4015ef:  48 63 d3  都不等      movslq %ebx,%rdx              rdx = ebx        1   2
  4015f2:  8b 7c 94 30           mov    0x30(%rsp,%rdx,4),%edi edi = (rsp+4×rdx+48)
  4015f6:  39 7c 84 30           cmp    %edi,0x30(%rsp,%rax,4) edi ≠ (rsp+4×rax+48)
  4015fa:  75 e8     且 ≤ 6      jne    4015e4 <phase_6+0x20>
  4015fc:  e8 a6 02 00 00        callq  4018a7 <explode_bomb>
  401601:  eb e1                 jmp    4015e4 <phase_6+0x20>
  401603:  44 89 e5      1       mov    %r12d,%ebp
  401606:  83 fd 05              cmp    $0x5,%ebp              ebp >5, jump   0
  401609:  7f 18                 jg     401623 <phase_6+0x5f>
  40160b:  48 63 c5              movslq %ebp,%rax       等于  rax = ebp   0 1
  40160e:  8b 44 84 30           mov    0x30(%rsp,%rax,4),%eax  eax = (rsp+4×rax+48)
  401612:  83 e8 01              sub    $0x1,%eax              eax --
  401615:  83 f8 05              cmp    $0x5,%eax              (eax >5, bomb.)    eax <= 6
  401618:  77 c3                 ja     4015dd <phase_6+0x19>
  40161a:  44 8d 65 01           lea    0x1(%rbp),%r12d        r12d = rbp+1     1   2
  40161e:  44 89 e3              mov    %r12d,%ebx             ebx = r12d       1   2
  401621:  eb c4                 jmp    4015e7 <phase_6+0x23>
  401623:  be 00 00 00 00        mov    $0x0,%esi              esi = 0
  401628:  eb 07                 jmp    401631 <phase_6+0x6d>
  40162a:  48 89 14 cc           mov    %rdx,(%rsp,%rcx,8)     (rsp+8rcx) = ~
  40162e:  83 c6 01              add    $0x1,%esi              esi++
  401631:  83 fe 05  Input[rcx] ≤ 1, edx   cmp    $0x5,%esi     esi >5, jump
  401634:  7f 1c  else   edx+8(Input-1)   jg     401652 <phase_6+0x8e>
  401636:  b8 01 00 00 00        mov    $0x1,%eax              eax = 1
  40163b:  ba d0 52 40 00        mov    $0x4052d0,%edx         edx = 0X4052d0
  401640:  48 63 ce              movslq %esi,%rcx              rcx = esi
  401643:  39 44 8c 30           cmp    %eax,0x30(%rsp,%rcx,4) (rsp+4rcx+48) ≤ eax
```



```
(rsp+4+(ax+16))<eax
401647:  7e e1                  jle    40162a <phase_6+0x66>
401649:  48 8b 52 08            mov    0x8(%rdx),%rdx      rdx += 8         >eax
40164d:  83 c0 01               add    $0x1,%eax           eax ++
401650:  eb ee                  jmp    401640 <phase_6+0x7c>
401652:  48 8b 1c 24            mov    (%rsp),%rbx         rbx = (rsp)
401656:  48 89 d9               mov    %rbx,%rcx           rcx = rbx
401659:  b8 01 00 00 00         mov    $0x1,%eax           eax = 1
40165e:  eb 11                  jmp    401671 <phase_6+0xad>
401660:  48 63 d0               movslq %eax,%rdx           rdx = eax
401663:  48 8b 14 d4            mov    (%rsp,%rdx,8),%rdx  rdx = (rsp+ 8 rdx)
401667:  48 89 51 08            mov    %rdx,0x8(%rcx)      (rcx+8) = rdx
40166b:  83 c0 01               add    $0x1,%eax           eax++
40166e:  48 89 d1               mov    %rdx,%rcx           rcx = rdx
401671:  83 f8 05               cmp    $0x5,%eax           eax ≤ 5 , jump
401674:  7e ea                  jle    401660 <phase_6+0x9c>
401676:  48 c7 41 08 00 00 00   movq   $0x0,0x8(%rcx)      (rcx+8) = 0
40167d:  00
40167e:  bd 00 00 00 00         mov    $0x0,%ebp           ebp = 0
401683:  eb 07                  jmp    40168c <phase_6+0xc8>
401685:  48 8b 5b 08            mov    0x8(%rbx),%rbx      rbx = (rbx + 8)
401689:  83 c5 01               add    $0x1,%ebp           ebp ++
40168c:  83 fd 04               cmp    $0x4,%ebp
40168f:  7f 11                  jg     4016a2 <phase_6+0xde>  ebp > 4 , return
401691:  48 8b 43 08            mov    0x8(%rbx),%rax      rax = (rbx+8)    ebp≤4
401695:  8b 00                  mov    (%rax),%eax         eax = (rax)
401697:  39 03                  cmp    %eax,(%rbx)         (rbx) >= eax , jump
401699:  7d ea                  jge    401685 <phase_6+0xc1>
40169b:  e8 07 02 00 00         callq  4018a7 <explode_bomb>  (rbx) < eax , bomb
4016a0:  eb e3                  jmp    401685 <phase_6+0xc1>
4016a2:  48 83 c4 50            add    $0x50,%rsp
4016a6:  5b                     pop    %rbx
4016a7:  5d                     pop    %rbp
4016a8:  41 5c                  pop    %r12
4016aa:  c3                     retq
```

链表降序排列.

2 , 1 , 6 , 5 , 4 , 3



```
0000000000401a38 <phase_defused>:
  401a38:  83 3d 2d 3d 00 00 06        cmpl   $0x6,0x3d2d(%rip)        # 40576c <num_input_strings>
  401a3f:  74 01                       je     401a42 <phase_defused+0xa>
  401a41:  c3                          retq
  401a42:  48 83 ec 68                 sub    $0x68,%rsp
  401a46:  4c 8d 44 24 10              lea    0x10(%rsp),%r8
  401a4b:  48 8d 4c 24 08              lea    0x8(%rsp),%rcx
  401a50:  48 8d 54 24 0c              lea    0xc(%rsp),%rdx
  401a55:  be 81 33 40 00              ~~mov    $0x403381,%esi~~
  401a5a:  bf 70 58 40 00              mov    $0x405870,%edi
  401a5f:  b8 00 00 00 00              mov    $0x0,%eax
  401a64:  e8 a7 f6 ff ff              callq  401110 <__isoc99_sscanf@plt>
  401a69:  83 f8 03                    cmp    $0x3,%eax
  401a6c:  74 0f                       je     401a7d <phase_defused+0x45>
  401a6e:  bf c0 32 40 00              mov    $0x4032c0,%edi
  401a73:  e8 e8 f5 ff ff              callq  401060 <puts@plt>
  401a78:  48 83 c4 68                 add    $0x68,%rsp
  401a7c:  c3                          retq
  401a7d:  be 8a 33 40 00              mov    $0x40338a,%esi
  401a82:  48 8d 7c 24 10              lea    0x10(%rsp),%rdi
  401a87:  e8 39 fd ff ff              callq  4017c5 <strings_not_equal>
  401a8c:  85 c0                       test   %eax,%eax
  401a8e:  75 de                       jne    401a6e <phase_defused+0x36>
  401a90:  bf 60 32 40 00              mov    $0x403260,%edi
  401a95:  e8 c6 f5 ff ff              callq  401060 <puts@plt>
  401a9a:  bf 88 32 40 00              mov    $0x403288,%edi
  401a9f:  e8 bc f5 ff ff              callq  401060 <puts@plt>
  401aa4:  b8 00 00 00 00              mov    $0x0,%eax
  401aa9:  e8 3a fc ff ff              callq  4016e8 <secret_phase>
  401aae:  eb be                       jmp    401a6e <phase_defused+0x36>
```

输入6个字符

"%d %d %s"

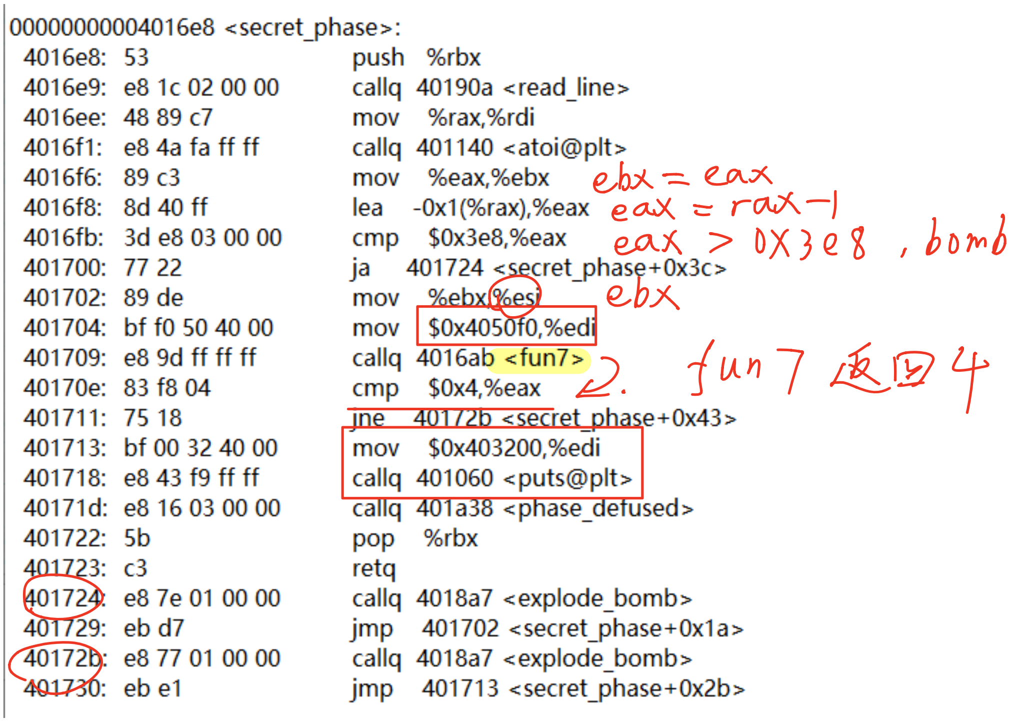
3个

输入3个内容.

DrEvil

eax ≠ 0

found, not succeed



```
0000000004016e8 <secret_phase>:
  4016e8:  53                   push   %rbx
  4016e9:  e8 1c 02 00 00       callq  40190a <read_line>
  4016ee:  48 89 c7             mov    %rax,%rdi
  4016f1:  e8 4a fa ff ff       callq  401140 <atoi@plt>
  4016f6:  89 c3                mov    %eax,%ebx
  4016f8:  8d 40 ff             lea    -0x1(%rax),%eax
  4016fb:  3d e8 03 00 00       cmp    $0x3e8,%eax
  401700:  77 22                ja     401724 <secret_phase+0x3c>
  401702:  89 de                mov    %ebx,%esi
  401704:  bf f0 50 40 00       mov    $0x4050f0,%edi
  401709:  e8 9d ff ff ff       callq  4016ab <fun7>
  40170e:  83 f8 04             cmp    $0x4,%eax
  401711:  75 18                ~~jne    40172b <secret_phase+0x43>~~
  401713:  bf 00 32 40 00       mov    $0x403200,%edi
  401718:  e8 43 f9 ff ff       callq  401060 <puts@plt>
  40171d:  e8 16 03 00 00       callq  401a38 <phase_defused>
  401722:  5b                   pop    %rbx
  401723:  c3                   retq
  401724:  e8 7e 01 00 00       callq  4018a7 <explode_bomb>
  401729:  eb d7                jmp    401702 <secret_phase+0x1a>
  40172b:  e8 77 01 00 00       callq  4018a7 <explode_bomb>
  401730:  eb e1                jmp    401713 <secret_phase+0x2b>
```

ebx = eax

eax = rax-1

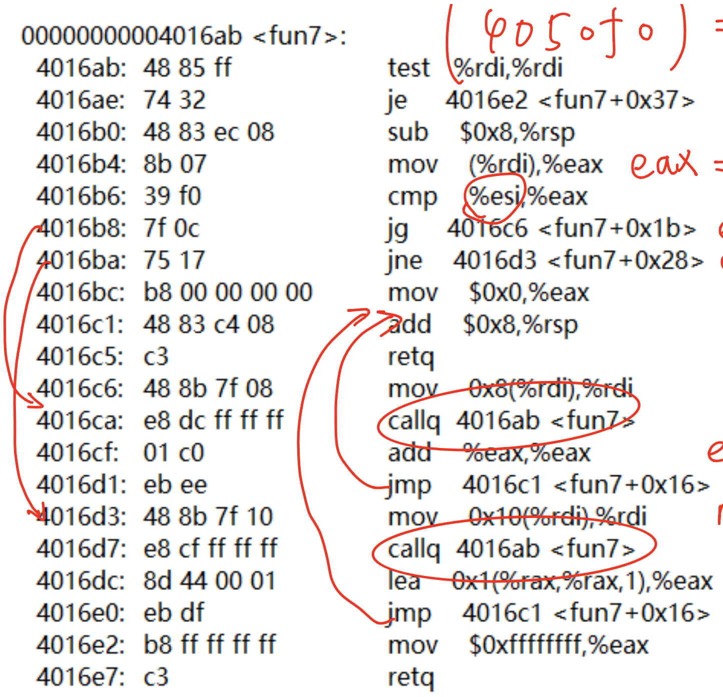
eax > 0x3e8 , bomb

ebx

2.

fun7 返回4

(405of0) =0X24



```
0000000004016ab <fun7>:
  4016ab:  48 85 ff              test   %rdi,%rdi
  4016ae:  74 32                 je     4016e2 <fun7+0x37>
  4016b0:  48 83 ec 08           sub    $0x8,%rsp
  4016b4:  8b 07                 mov    (%rdi),%eax
  4016b6:  39 f0                 cmp    %esi,%eax
  4016b8:  7f 0c                 jg     4016c6 <fun7+0x1b>
  4016ba:  75 17                 jne    4016d3 <fun7+0x28>
  4016bc:  b8 00 00 00 00        mov    $0x0,%eax
  4016c1:  48 83 c4 08           add    $0x8,%rsp
  4016c5:  c3                    retq
  4016c6:  48 8b 7f 08           mov    ~~0x8(%rdi),%rdi~~
  4016ca:  e8 dc ff ff ff        callq  4016ab <fun7>
  4016cf:  01 c0                 add    %eax,%eax
  4016d1:  eb ee                 jmp    4016c1 <fun7+0x16>
  4016d3:  48 8b 7f 10           mov    ~~0x10(%rdi),%rdi~~
  4016d7:  e8 cf ff ff ff        callq  4016ab <fun7>
  4016dc:  8d 44 00 01           lea    ~~0x1(%rax,%rax,1)~~,%eax
  4016e0:  eb df                 jmp    4016c1 <fun7+0x16>
  4016e2:  b8 ff ff ff ff        mov    $0xffffffff,%eax
  4016e7:  c3                    retq
```

rdi = 0 , return -1,

eax = 0X24

eax > esi

eax < esi

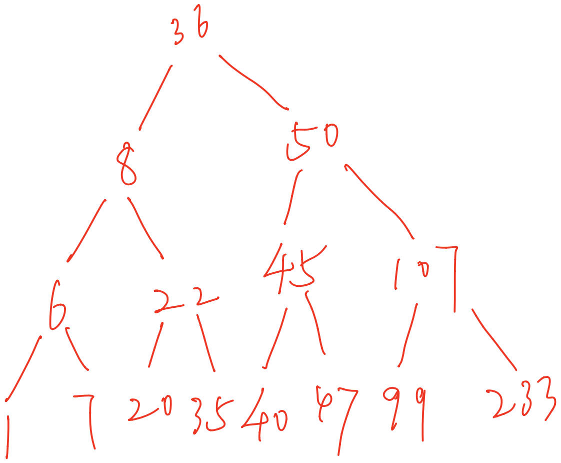
rdi = rdi + 8

eax = 2eax

rdi = rdi + 16

eax = 2rax + 1

(0+1) × 2 × 2



7