```
0x16f
--------
$s2 = 0xffffffff & c[0x4]
assert(0 == (0xff & (s[0x0] >> 0xa0)))
$s3 = intcall0($s2, 0x400)
assert($s3)
m[0x0] = 0xffffffff & $s2
m[0x20] = 0x2
assert(s[sha3(0x0, 0x40)] <= msg.value)
$s4 = s[0x1]
$s5 = 0x1 + $s4
$s9 = s[0x1]
s[0x1] = $s5
if (0 == $s9 <= $s5){
  m[0x0] = 0x1
  $s10 = sha3(0x0, 0x20)
  $t = $s10
  $s10 = $s10 + $s9
  $s12 = $s5 + $t
  while (0x1) {
    if ($s10 <= $s12)
        break
    s[$s12] = 0xffffffffffffffff0000000000000000000000000000000000000000000000000000 & s[$s12]
    $s12 = 0x1 + $s12
  }

  goto 0x747
}
m[0x0] = 0x1
$s5 = sha3(0x0, 0x20)
$s6 = $m
$m = 0x40 + $m
m[$s6] = msg.sender
m[0x20 + $s6] = 0xffffffff & $s2
$t = $s4 + $s5
$s5 = $t
s[$s5] = (ad_mask & m[$s6]) | (0xffffffffffffffffffffffff0000000000000000000000000000000000000000 & s[$t])
s[$s5] = ((0xffffffff & m[0x20 + $s6]) << 0xa0) | (0xffffffffffffffff00000000ffffffffffffffffffffffffffffffffffffffff & s[$s5])
$s5 = 0xffffffff & $s2
m[0x0] = $s5
m[0x20] = 0x3
$s7 = sha3(0x0, 0x40)
s[$s7] = s[$s7] - 0x1
log3($m, 0x0, 0x1395824e255e4df6983f1f2d84bb8afe054b8c65b520192076cf671c14068f88, msg.sender, $s5)
stop()
```