

```
0x546
-----
assert(0 == msg.value)
$s3 = calldatasize - 0x4
assert($s3 >= 0x40)
$s2 = 0x4 + $s3
$t = c[0x4]
assert($t <= 0x100000000)
$s5 = 0x4 + $t
assert((0x20 + $s5) <= $s2)
$t = $s5
$s5 = c[$s5]
$s6 = 0x20 + $t
assert(0 == (($s5 > 0x100000000) | (($s6 + $s5) > $s2)))
$t = $s5
$s5 = $s6
$s6 = $t
$s8 = $m
$m = $m + (0x20 + (0x20 * ((0x1f + $t) / 0x20)))
m[$s8] = $t
$s9 = 0x20 + $s8
calldatacopy($s9, $s5, $s6)
m[$s9 + $s6] = 0x0
$s2 = $s8
$s3 = c[0x24]
m[0x0] = msg.sender
m[0x20] = 0x9
assert(0 == (0x0 == s[sha3(0x0, 0x40)]))
m[0x0] = msg.sender
m[0x20] = 0xa
$s4 = sha3(0x0, 0x40)
$s7 = 0x1 + s[$s4]
s[$s4] = $s7
m[0x0] = $s4
s[sha3(0x0, 0x20) + ($s7 - 0x1)] = $s3
m[$m] = msg.sender
$s9 = 0x20 + $m
$s10 = 0x20 + $s9
m[$s10] = $s3
$s10 = 0x20 + $s10
m[$s9] = $s10 - $m
m[$s10] = m[$s2]
$s10 = 0x20 + $s10
$t = m[$s2]
$s11 = 0x20 + $s2
$s16 = 0x0
while (0x1) {
    if ($s16 >= $t)
        break
    m[$s10 + $s16] = m[$s11 + $s16]
    $s16 = 0x20 + $s16
}
$s11 = $t
$t = $s10
$s10 = $s11
$s11 = $s11 + $t
$t = $s10
$s10 = $s11
$s11 = 0x1f & $t
if ($s11){
    $s12 = $s10 - $s11
    m[$s12] = (! ((0x100 ** (0x20 - $s11)) - 0x1)) & m[$s12]
    $s10 = 0x20 + $s12
}
log1($m, $s10 - $m, 0x1f5802dd7d382bf95bfb5c6a036986b42ca5c78d31394262c2b73ceb55bd5f30)
m[0x0] = msg.sender
m[0x20] = 0xa
if (s[sha3(0x0, 0x40)] == s[0x5]){
    m[0x0] = msg.sender
    m[0x20] = 0xa
    $s5 = sha3(0x0, 0x40)
    $s8 = s[$s5]
    s[$s5] = 0x0
    if ($s8 > 0x0){
        m[0x0] = $s5
        $s10 = sha3(0x0, 0x20)
        $s8 = intcall1($s10, $s10 + $s8, 0xfd0)

    }
}
}
stop()
```