```
0x35f
assert(0 == msg.value)
$s2 = c[0x4]
$s4 = c[0x24]
$s3 = 0x24 + $s4
$s4 = c[0x4 + $s4]
$s5 = intcall0(0xc95)
\$s7 = \$m
m = 0x80 + m
$s8 = 0x4
$s9 = $s7
 0x2fc8
m[\$s9] = 0x0
 $s8 = $s8 - 0x1
 $s9 = 0x20 + $s9
 if ($s8) goto 0x2fc8
```

```
0x2fdd
assert(ad mask & s[0xd])
$s8 = ad mask \& s[0xd]
m[0xa0 + $m] = 0x0
m[0x4 + $m] = $s2
m[0x24 + $m] = 0x40
m[0x44 + $m] = $s4
$s15 = 0x64 + $m
calldatacopy($s15, $s3, $s4)
assert(extcodesize($s8))
assert(call(msg.gas - 0x2c6, $s8, 0x0, $m, ($s15 + $s4) - $m, $m, 0xa0))
$s8 = $m
$s9 = 0x80 + $m
m = 0x20 + s9
$t = $s8
$s8 = m[$s9]
$s11 = intcall0(0x252c)
$s12 = intcall0(0x2534)
$s16 = $m
if (msize() >= $m){
 $s16 = msize()
m[\$s16] = \$s8
$s15 = $s16
$s16 = 0x20 + $s16
$s17 = $t
$s18 = $s8
while (0x1) {
 if (\$s18 < 0x20)
       break
 m[\$s16] = m[\$s17]
 $s16 = 0x20 + $s16
 $s17 = 0x20 + $s17
 $s18 = $s18 - 0x20
$s19 = (0x100 ** (0x20 - $s18)) - 0x1
m[\$s16] = (m[\$s16] \& \$s19) | (m[\$s17] \& (! \$s19))
m[\$m] = 0 \times 20
$s4 = 0x20 + $m
m[\$s4] = m[\$s15]
$s4 = 0x20 + $s4
t = m[$s15]
$s5 = 0x20 + $s15
$s10 = 0x0
while (0x1) {
 if (\$s10 >= \$t)
       break
 m[\$s10 + \$s4] = m[\$s5 + \$s10]
 $s10 = 0x20 + $s10
$s5 = $t
$t = $s4
$s4 = $s5
$s5 = $s5 + $t
$t = $s4
$s4 = $s5
$s5 = 0x1f \& $t
if ($s5){
 $s6 = $s4 - $s5
 m[\$s6] = (! ((0x100 ** (0x20 - \$s5)) - 0x1)) \& m[\$s6]
 $s4 = 0x20 + $s6
return($m, $s4 - $m)
```