```
0x176
--------
assert(0 == msg.value)
$s2 = ad_mask & c[0x4]
$s3 = ad_mask & c[0x24]
$s15 = ad_mask & $s2
m[0x20 + $m] = 0x0
m[$m] = 0x870c426d << 0xe0
assert(extcodesize($s15))
assert(call(msg.gas - 0x2c6, $s15, 0x0, $m, (0x4 + $m) - $m, $m, 0x20))
$s14 = m[$m]
$s15 = ad_mask & s[0x0]
m[0x20 + $m] = 0x0
m[$m] = 0x4e94c829 << 0xe0
assert(extcodesize($s15))
assert(call(msg.gas - 0x2c6, $s15, 0x0, $m, (0x4 + $m) - $m, $m, 0x20))
$s15 = ad_mask & m[$m]
m[0x20 + $m] = 0x0
m[$m] = 0x8cfb8f2100000000000000000000000000000000000000000000000000000000
m[0x4 + $m] = ad_mask & $s14
assert(extcodesize($s15))
assert(call(msg.gas - 0x2c6, $s15, 0x0, $m, (0x24 + $m) - $m, $m, 0x20))
assert(m[$m])
$s15 = ad_mask & $s14
m[0x20 + $m] = 0x0
m[$m] = 0x9f7e1bf600000000000000000000000000000000000000000000000000000000
m[0x4 + $m] = ad_mask & $s2
assert(extcodesize($s15))
assert(call(msg.gas - 0x2c6, $s15, 0x0, $m, (0x24 + $m) - $m, $m, 0x20))
assert(m[$m])
$s15 = ad_mask & s[0x0]
m[0x20 + $m] = 0x0
m[$m] = 0x22763ae1 << 0xe0
assert(extcodesize($s15))
assert(call(msg.gas - 0x2c6, $s15, 0x0, $m, (0x4 + $m) - $m, $m, 0x20))
assert(m[$m])
assert(0 == (0xff & (s[0x0] >> 0xa0)))
s[0x0] = 0x1000000000000000000000000000000000000000 | (0xffffffffffffffffffffff00ffffffffffffffffffffffffffffffffffffffff & s[0x0])
$s17 = ad_mask & $s2
m[0x20 + $m] = 0x0
m[$m] = 0xa0e16fed << 0xe0
assert(extcodesize($s17))
assert(call(msg.gas - 0x2c6, $s17, 0x0, $m, (0x4 + $m) - $m, $m, 0x20))
$s15 = intcall1(0x3f480, m[$m], 0x6da)
$s16 = ad_mask & s[0x0]
m[0x20 + $m] = 0x0
m[$m] = 0x188ec356 << 0xe0
assert(extcodesize($s16))
assert(call(msg.gas - 0x2c6, $s16, 0x0, $m, (0x4 + $m) - $m, $m, 0x20))
assert(m[$m] > $s15)
$s15 = ad_mask & $s2
m[0x20 + $m] = 0x0
m[$m] = 0xdf2a29da << 0xe0
assert(extcodesize($s15))
assert(call(msg.gas - 0x2c6, $s15, 0x0, $m, (0x4 + $m) - $m, $m, 0x20))
$s5 = m[$m]
$s6 = 0x0
while (0x1) {
  $s15 = ad_mask & $s2
  m[0x20 + $m] = 0x0
  m[$m] = 0x27ce5b8c << 0xe0
  assert(extcodesize($s15))
  assert(call(msg.gas - 0x2c6, $s15, 0x0, $m, (0x4 + $m) - $m, $m, 0x20))
  if ($s6 >= m[$m])
      break
  $s15 = ad_mask & $s2
  m[0x20 + $m] = 0x0
  m[$m] = 0x65957bf500000000000000000000000000000000000000000000000000000000
  m[0x4 + $m] = $s6
  assert(extcodesize($s15))
  assert(call(msg.gas - 0x2c6, $s15, 0x0, $m, (0x24 + $m) - $m, $m, 0x20))
  $s7 = m[$m]
  $s15 = ad_mask & $s7
  m[0x20 + $m] = 0x0
  m[$m] = 0x70a0823100000000000000000000000000000000000000000000000000000000
  m[0x4 + $m] = ad_mask & $s3
  assert(extcodesize($s15))
  assert(call(msg.gas - 0x2c6, $s15, 0x0, $m, (0x24 + $m) - $m, $m, 0x20))
  $s8 = m[$m]
  $s15,$s16,$s17,$s18 = intcall4($s8, $s6, $s2, 0x903)
  $s11 = $s17
  $s12 = $s18
  if ($s8 > 0x0){
    $s15 = ad_mask & $s7
    m[0x20 + $m] = 0x0
    m[$m] = 0xd333d7cf0000000000000000000000000000000000000000000000000000000000000000
    m[0x4 + $m] = ad_mask & $s3
    m[0x24 + $m] = $s8
    assert(extcodesize($s15))
    assert(call(msg.gas - 0x2c6, $s15, 0x0, $m, (0x44 + $m) - $m, $m, 0x20))
    $s15 = m[$m]
    $s22 = ad_mask & s[0x0]
    m[0x20 + $m] = 0x0
    m[$m] = 0x4e94c829 << 0xe0
    assert(extcodesize($s22))
    assert(call(msg.gas - 0x2c6, $s22, 0x0, $m, (0x4 + $m) - $m, $m, 0x20))
    $s22 = ad_mask & m[$m]
    $s24 = ad_mask & $s2
    m[0x20 + $m] = 0x0
    m[$m] = 0x870c426d << 0xe0
    assert(extcodesize($s24))
    assert(call(msg.gas - 0x2c6, $s24, 0x0, $m, (0x4 + $m) - $m, $m, 0x20))
    $s24 = m[$m]
    $s30 = intcall1($s16, balance(ad_mask & $s3), 0xf64)
    m[0x20 + $m] = 0x0
    m[$m] = 0x6051fa2c0000000000000000000000000000000000000000000000000000000000000000
    m[0x4 + $m] = ad_mask & $s24
    m[0x24 + $m] = ad_mask & $s7
    m[0x44 + $m] = ad_mask & $s3
    m[0x64 + $m] = ad_mask & $s2
    m[0x84 + $m] = $s8
    m[0xa4 + $m] = $s16
    m[0xc4 + $m] = $s30
    assert(extcodesize($s22))
    assert(call(msg.gas - 0x2c6, $s22, 0x0, $m, (0xe4 + $m) - $m, $m, 0x20))
    $s23 = m[$m]
  }
  if ($s16 > 0x0){
    $s15 = ad_mask & $s5
    m[0x20 + $m] = 0x0
    m[$m] = 0x23b872dd0000000000000000000000000000000000000000000000000000000000000000
    m[0x4 + $m] = ad_mask & $s2
    m[0x24 + $m] = self
    m[0x44 + $m] = $s16
    assert(extcodesize($s15))
    assert(call(msg.gas - 0x2c6, $s15, 0x0, $m, (0x64 + $m) - $m, $m, 0x20))
    assert(m[$m])
    $s15 = ad_mask & $s5
    m[0x20 + $m] = 0x0
    m[$m] = 0x1baffe3800000000000000000000000000000000000000000000000000000000
    m[0x4 + $m] = ad_mask & $s3
    m[0x24 + $m] = $s16
    assert(extcodesize($s15))
    assert(call(msg.gas - 0x2c6, $s15, 0x0, $m, (0x44 + $m) - $m, $m, 0x20))
    $s16 = m[$m]
  }
  if ($s11 > 0x0){
    $s15 = ad_mask & $s5
    $s18 = ad_mask & $s2
    m[0x20 + $m] = 0x0
    m[$m] = 0xed23378b << 0xe0
    assert(extcodesize($s18))
    assert(call(msg.gas - 0x2c6, $s18, 0x0, $m, (0x4 + $m) - $m, $m, 0x20))
    $s18 = m[$m]
    m[0x20 + $m] = 0x0
    m[$m] = 0x23b872dd0000000000000000000000000000000000000000000000000000000000000000
    m[0x4 + $m] = ad_mask & $s2
    m[0x24 + $m] = ad_mask & $s18
    m[0x44 + $m] = $s11
    assert(extcodesize($s15))
    assert(call(msg.gas - 0x2c6, $s15, 0x0, $m, (0x64 + $m) - $m, $m, 0x20))
    assert(m[$m])
  }
  if ($s12 > 0x0){
    $s15 = ad_mask & $s5
    $s18 = ad_mask & $s2
    m[0x20 + $m] = 0x0
    m[$m] = 0x870c426d << 0xe0
    assert(extcodesize($s18))
    assert(call(msg.gas - 0x2c6, $s18, 0x0, $m, (0x4 + $m) - $m, $m, 0x20))
    $s18 = ad_mask & m[$m]
    m[0x20 + $m] = 0x0
    m[$m] = 0xcc8c9af << 0xe0
    assert(extcodesize($s18))
    assert(call(msg.gas - 0x2c6, $s18, 0x0, $m, (0x4 + $m) - $m, $m, 0x20))
    $s18 = m[$m]
    m[0x20 + $m] = 0x0
    m[$m] = 0x23b872dd0000000000000000000000000000000000000000000000000000000000000000
    m[0x4 + $m] = ad_mask & $s2
    m[0x24 + $m] = ad_mask & $s18
    m[0x44 + $m] = $s12
    assert(extcodesize($s15))
    assert(call(msg.gas - 0x2c6, $s15, 0x0, $m, (0x64 + $m) - $m, $m, 0x20))
    assert(m[$m])
  }
  $s6 = 0x1 + $s6
}
$s15 = ad_mask & $s2
m[0x20 + $m] = 0x0
m[$m] = 0xa0695f24 << 0xe0
assert(extcodesize($s15))
assert(call(msg.gas - 0x2c6, $s15, 0x0, $m, (0x4 + $m) - $m, $m, 0x20))
$s16 = m[$m]
s[0x0] = 0xffffffffffffffffffffff00ffffffffffffffffffffffffffffffffffffffff & s[0x0]
m[$m] = 0x1
return($m, (0x20 + $m) - $m)
```