```
0x1fd
--------
assert(0 == msg.value)
$s2 = c[0x4]
$s4 = intcall2(0x8e1)
assert($s2 < $s4)
$s4 = intcall3(0x8f4)
assert($s4)
$s4 = intcall8(msg.sender, 0x908)
assert($s4)
$s4 = intcall0(msg.sender, 0x91c)
assert(0 == $s4)
$s4 = intcall5($s2, 0x92f)
assert($s4)
m[0x0] = msg.sender
m[0x20] = 0x1
$s3 = sha3(0x0, 0x40)
$s4 = intcall9(0x95b)
s[$s3] = $s4
$s5 = 0x1 + $s3
$s6 = s[$s5]
$t = $s5
$s5 = $s6
$s4 = $t
$s6 = 0x1 + $s6
$s10 = s[$t]
s[$t] = $s6
if (0 == $s10 <= $s6){
  m[0x0] = $t
  $s11 = sha3(0x0, 0x20)
  $s10 = intcall1($s6 + $s11, $s11 + $s10, 0x1475)
}

m[0x0] = $s4
s[$s5 + sha3(0x0, 0x20)] = $s2
assert($s2 < s[0x2])
m[0x0] = 0x2
$s5 = 0x3 + ((0x4 * $s2) + sha3(0x0, 0x20))
s[$s5] = 0x1 + s[$s5]
s[0x5] = 0x1 + s[0x5]
stop()
```