```
──────────────────── MODULE hmac ────────────────────
EXTENDS Integers, Sequences, FiniteSets, TLC, Bitwise

VARIABLES message, key, sentHash, processedHash, equalityCheck, A, B, C, D, AA, BB, CC, DD, M, K

BLOCK_SIZE ≜ 64

RECURSIVE shiftL(_, _)
shiftL(n, pos) ≜
   IF pos = 0
    THEN n
    ELSE LET double(z) ≜ 2 ∗ z
          IN   shiftL(double(n), pos − 1)

LeftRotate(x, c) ≜
   (shiftL(x, c) | shiftR(x, 32 − c))%(2^32)


MD5(m) ≜
     LET
         ProcessChunk(chunk) ≜
           LET P ≜ [j ∈ 0 . . 15 ↦ SubSeq(m, (chunk − 1) ∗ 512 + j ∗ 32 + 1, (chunk − 1) ∗ 512 + (j +
           IN
               ∧ AA′ = A
               ∧ BB′ = B
               ∧ CC′ = C
               ∧ DD′ = D
               ∧ ∀ i ∈ 0 . . 63 :
                  LET
                     F ≜ IF i ∈ 0 . . 15 THEN (B & C) | ((Not(B)) & D)
                          ELSE IF i ∈ 16 . . 31 THEN (D & B) | ((Not(D)) & C)
                          ELSE IF i ∈ 32 . . 47 THEN ((B^^C)^^D)
                          ELSE C^^(B | (Not(D)))
                     g ≜ IF i ∈ 0 . . 15 THEN i
                            ELSE IF i ∈ 16 . . 31 THEN (5 ∗ i + 1)%16
                            ELSE IF i ∈ 32 . . 47 THEN (3 ∗ i + 5)%16
                            ELSE (7 ∗ i)%16
                  IN
                     ∧ F′ = F + A + K[i] + P[g]
                     ∧ A′ = D
                     ∧ D′ = C
                     ∧ C′ = B
                     ∧ B′ = B + LeftRotate(F′, S[i])
               ∧ A′ = A + AA
               ∧ B′ = B + BB
               ∧ C′ = C + CC
               ∧ D′ = D + DD
```

```
        digest ≜ ⟨A, B, C, D⟩
    IN
        digest

ExtendedKey ≜ IF Len(key) > BLOCK_SIZE THEN MD5(key) ELSE  Append(key, ⟨0⟩^(BLOCK_SIZE−Len

ipad  ≜ [i ∈ 1 .. Len(ExtendedKey) ↦ ExtendedKey[i]^^54]
opad  ≜ [i ∈ 1 .. Len(ExtendedKey) ↦ ExtendedKey[i]^^92]

HashFunction(m, k) ≜
    LET
        innerHash ≜ MD5(ipad ∘ m)
        resultHash ≜ MD5(opad ∘ innerHash)
    IN
        resultHash

SendHash ≜
    ∧ sentHash′ = HashFunction(message, ExtendedKey)
    ∧ UNCHANGED ⟨message, key, processedHash, equalityCheck, A, B, C, D, AA, BB, CC, DD, K, S, M

ProcessHash ≜
    ∧ processedHash′ = HashFunction(message, ExtendedKey)
    ∧ UNCHANGED ⟨message, key, sentHash, equalityCheck, A, B, C, D, AA, BB, CC, DD, K, S, M⟩

CompareHashes ≜
    ∧ equalityCheck′ = (sentHash = processedHash)
    ∧ UNCHANGED ⟨message, key, sentHash, processedHash, A, B, C, D, AA, BB, CC, DD, K, S, M⟩

Init ≜
    ∧ message = ⟨0, 6, 56, 78, 87, 12⟩
    ∧ key = ⟨0, 6, 56, 78, 87, 12⟩
    ∧ sentHash = ⟨⟩
    ∧ processedHash = ⟨⟩
    ∧ equalityCheck  = FALSE
    ∧ A  = 13
    ∧ B  = 17
    ∧ C  = 19
    ∧ D  = 23
    ∧ K = ⟨
            102, 205, 307, 410,
            512, 615, 718, 820,
            923, 1025, 1128, 1230,
            1333, 1435, 1538, 1640,
            1743, 1845, 1948, 2050,
            2153, 2255, 2358, 2460,
            2563, 2665, 2768, 2870,
            2973, 3075, 3178, 3280,
            3383, 3485, 3588, 3690,
            3793, 3895, 3998, 4100,
            4203, 4305, 4408, 4510,
            4613, 4715, 4818, 4920,
            5023, 5125, 5228, 5330,
            5433, 5535, 5638, 5740,
            5843, 5945, 6048, 6150,
            6253, 6355, 6458, 6560
        ⟩

    ∧ S = ⟨
        7, 12, 17, 22,
        7, 12, 17, 22,
        7, 12, 17, 22,
        7, 12, 17, 22,
        5, 9, 14, 20,
        5, 9, 14, 20,
        5, 9, 14, 20,
        5, 9, 14, 20,
        4, 11, 16, 23,
        4, 11, 16, 23,
        4, 11, 16, 23,
        4, 11, 16, 23,
        6, 10, 15, 21,
        6, 10, 15, 21,
        6, 10, 15, 21,
        6, 10, 15, 21
        ⟩
    ∧ AA = A
    ∧ BB = B
    ∧ CC = C
    ∧ DD = D
    ∧ M = ⟨0, 6, 56, 78, 87, 12⟩

Next ≜
    ∨ SendHash
    ∨ ProcessHash
    ∨ CompareHashes

Spec ≜
    Init ∧ □[Next]_⟨message, key, sentHash, processedHash, equalityCheck, A, B, C, D, AA, BB, CC, DD, M, K, S⟩
```