CRYPTOGRAPHY #04

SHA - 2

Jacek Tchórzewski, jacek.tchorzewski@pk.edu.pl

1. Hashing Functions

Hashing Functions are One – Way functions:

$$\forall x \in X,\ f\colon x \to y \wedge \sim(\exists g\colon y \to x)$$

Which means, that for every argument exists one, and only one value, however, there is no reverse function (which can transform value back to the original argument). There is an additional condition which One - Way function must meet to become a hashing function *h*:

$$\mathrm{h}\colon \{0,\ 1\}^{*} \to \{0,\ 1\}^{n},\ n \geq 1$$

Which means that as an argument (input of a hashing function) binary strings of any length are considered and outputs are binary strings with a fixed length.

Parameters of Hashing Functions:

- The same message will return always the same hash.
- Hash Function does not give any constraint on input data size.
- Output hash has constant length.
- Output hash should be easy to compute.
- Given *h* and *h(x)*, it is computationally infeasible to determine *x* - **preimage resistance**.
- Given *h* and *x*, it is computationally infeasible to find an *x'* != *x* such that *h(x)* = *h(x')* - **second preimage resistance**.
- It should be hard to find any random *x* and *y* such that *h(x)* = *h(y)* – **collision resistance**.

Hashing Functions can be used for:

- Digital Signatures.
- Indexing data (Hash Tables).
- Secure passwords storage.
- etc.

2. SHA

SHA stands for Secure Hash Algorithm. National Institute of Standards and Technology is responsible for certification of Hashing Functions. There were 4 standards (also called SHSs – Secure Hash Standards): SHA-0 (for a short time), SHA-1, SHA-2, SHA-3, however, nowadays only 2 of them (SHA-2 and SHA-3) are allowed to be used. SHA-2 and SHA-3 are not only Hashing Functions but rather families of Hashing Functions. For example: SHA – 2 contains SHA – 256 (which returns 256 bits hash), SHA – 512 (which is returning 512 bits hash), and more. For more information look at the : NIST FIPS PUB 180-4.

3. SHA – 2 additional functions:

- *RotateRight(w, n)* - Right binary rotation of binary word *w* by *n* positions.
- *ShiftRight(w, n)* - Right binary shift of binary word *w* by *n* positions.
- *bytearray(a, b)* - Function which converts element *a* into byte array of size *b*.
- *not(a)* - Negation of *a*.
- *len(a)* - Number of elements in vector *a.*
- *parseUint32(a)* - Function which is parsing byte array *a* into unsigned 32 bits integer.
- *a.append(b)* - Appends element *b* to the vector *a*. If *b* is also a vector, all elements of *b* are added to the vector *a*, and the vector *a* is containing *len(a) + len(b)* elements.
- $S(x, a, b, c) = RotateRight(x, a) \oplus RotateRight(x, b) \oplus RotateRight(x, c)$
- $s(x, a, b, c) = RotateRight(x, a) \oplus RotateRight(x, b) \oplus ShiftRight(x, c)$
- $Maj(a, b, c) = (a \wedge b) \oplus (a \wedge c) \oplus (b \wedge c)$
- $Ch(a, b, c) = (a \wedge b) \oplus (not(a) \wedge c)$

4. SHA-2 Padding Algorithm

m - byte array representing a message
n - block size (in bytes)
Step 1: pad(m, n):
Step 2: tmp = len(m) * 8
Step 3: m.append(0x80)
Step 4: i = len(m)
Step 5: while: *i* mod *n* != *n* – 8 do Steps 6, 7
Step 6: m.append(0x00)
Step 7: i = i + 1
Step 8: m.append(bytearray(tmp, 8))
Step 9: return *m*

5. SHA – 256 Algorithm

The hash is calculated from byte array *M*, *H* are initial hash constants and *K* are fixed constants given by NIST. *K* and *H* values are 32-bit unsigned integers given in Hex.

Step 1: H = (0x6a09e667, 0xbb67ae85, 0x3c6ef372, 0xa54ff53a, 0x510e527f,0x9b05688c, 0x1f83d9ab, 0x5be0cd19)

Step 2: K = (0x428a2f98, 0x71374491, 0xb5c0fbcf, 0xe9b5dba5, 0x3956c25b, 0x59f111f1, 0x923f82a4, 0xab1c5ed5, 0xd807aa98, 0x12835b01, 0x243185be, 0x550c7dc3, 0x72be5d74, 0x80deb1fe, 0x9bdc06a7, 0xc19bf174, 0xe49b69c1, 0xefbe4786, 0x0fc19dc6, 0x240ca1cc, 0x2de92c6f, 0x4a7484aa, 0x5cb0a9dc, 0x76f988da, 0x983e5152, 0xa831c66d, 0xb00327c8, 0xbf597fc7, 0xc6e00bf3, 0xd5a79147, 0x06ca6351, 0x14292967, 0x27b70a85, 0x2e1b2138, 0x4d2c6dfc, 0x53380d13, 0x650a7354, 0x766a0abb, 0x81c2c92e, 0x92722c85, 0xa2bfe8a1, 0xa81a664b, 0xc24b8b70, 0xc76c51a3, 0xd192e819, 0xd6990624, 0xf40e3585, 0x106aa070, 0x19a4c116, 0x1e376c08, 0x2748774c, 0x34b0bcb5, 0x391c0cb3, 0x4ed8aa4a, 0x5b9cca4f, 0x682e6ff3, 0x748f82ee, 0x78a5636f, 0x84c87814, 0x8cc70208, 0x90befffa, 0xa4506ceb, 0xbef9a3f7, 0xc67178f2)

Step 3: M = pad(M, 64)
Step 4: i = 0
Step 5: tmp = (0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00)
Step 6: while i < len(M) do Steps 7-23
Step 7: W = () , **note** W can be declared as 64B array
Step 8: for j = 0,...,15 do Step 9
Step 9: $W.append(parseUint32(M(\left\lfloor \frac{i}{64} \right\rfloor * 64 + j * 4, \left\lfloor \frac{i}{64} \right\rfloor * 64 + j * 4 + 3)))$.
Step 10: for j = 16,...,63 do Step 11
Step 11: W.append(W(j−16)+W(j−7)+s(W(j−15),7,18,3)+s(W(j−2),17,19,10))
Step 12: for j = 0,...,7 do Step 13
Step 13: tmp(j) = H(j)
Step 14: for j = 0,...,63 do Step 15-20
Step 15: t1 = K(j)+W(j)+S(tmp(4),6,11,25)+Ch(tmp(4),tmp(5),tmp(6))+tmp(7)
Step 16: t2 = Maj(tmp(0),tmp(1),tmp(2)) +S(tmp(0),2,13,22)
Step 17: for k = 7,...,1 do Step 18
Step 18: tmp(k) = tmp(k −1)
Step 19: tmp(0) = t1+t2
Step 20: tmp(4) = tmp(4) +t1
Step 21: for j = 0,...,7 do Step 22
Step 22: H(j) = H(j) +tmp(j)
Step 23: i = i+64
Step 24: return H(0)||H(1)||H(2)||H(3)||H(4)||H(5)||H(6)||H(7)

SHA-256 example values (padding + step 8 – 11):
message = „aaa”;

messageInHex = 0x616161

PaddedMessageInHex=
0x6161618000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000018

Iteration 1, W =
1633771904

Iteration 2, W =
1633771904, 0

Iteration 3, W =
1633771904, 0, 0

Iteration 4, W =
1633771904, 0, 0, 0

Iteration 5, W =
1633771904, 0, 0, 0, 0

Iteration 6, W =
1633771904, 0, 0, 0, 0, 0

Iteration 7, W =
1633771904, 0, 0, 0, 0, 0, 0

Iteration 8, W =
1633771904, 0, 0, 0, 0, 0, 0, 0

Iteration 9, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0

Iteration 10, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0

Iteration 11, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

Iteration 12, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

Iteration 13, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

Iteration 14, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

Iteration 15, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

Iteration 16, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24

Iteration 17, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904

Iteration 18, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040

Iteration 19, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452

Iteration 20, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702

Iteration 21, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224

Iteration 22, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944

Iteration 23, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916

Iteration 24, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330

Iteration 25, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234

Iteration 26, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088

Iteration 27, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438

Iteration 28, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355

Iteration 29, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638

Iteration 30, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429

Iteration 31, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674

Iteration 32, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742

Iteration 33, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307

Iteration 34, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862

Iteration 35, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684

Iteration 36, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431

Iteration 37, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981

Iteration 38, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098

Iteration 39, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945

Iteration 40, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674

Iteration 41, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462

Iteration 42, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317

Iteration 43, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049

Iteration 44, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245

Iteration 45, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359

Iteration 46, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958

Iteration 47, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858

Iteration 48, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184

Iteration 49, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992

Iteration 50, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992, 51922169

Iteration 51, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992, 51922169, 1293266598

Iteration 52, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992, 51922169, 1293266598, 1497201556

Iteration 53, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992, 51922169, 1293266598, 1497201556, 1084406760

Iteration 54, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992, 51922169, 1293266598, 1497201556, 1084406760, -1865537359

Iteration 55, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992, 51922169, 1293266598, 1497201556, 1084406760, -1865537359, 782543434

Iteration 56, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992,

51922169, 1293266598, 1497201556, 1084406760, -1865537359, 782543434, 731454045

Iteration 57, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992, 51922169, 1293266598, 1497201556, 1084406760, -1865537359, 782543434, 731454045, -156312910

Iteration 58, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992, 51922169, 1293266598, 1497201556, 1084406760, -1865537359, 782543434, 731454045, -156312910, 1962396194

Iteration 59, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992, 51922169, 1293266598, 1497201556, 1084406760, -1865537359, 782543434, 731454045, -156312910, 1962396194, 256089834

Iteration 60, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992, 51922169, 1293266598, 1497201556, 1084406760, -1865537359, 782543434, 731454045, -156312910, 1962396194, 256089834, 15995728

Iteration 61, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992, 51922169, 1293266598, 1497201556, 1084406760, -1865537359, 782543434, 731454045, -156312910, 1962396194, 256089834, 15995728, -741561803

Iteration 62, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992, 51922169, 1293266598, 1497201556, 1084406760, -1865537359, 782543434, 731454045, -156312910, 1962396194, 256089834, 15995728, -741561803, 1105195401

Iteration 63, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992, 51922169, 1293266598, 1497201556, 1084406760, -1865537359, 782543434, 731454045, -156312910, 1962396194, 256089834, 15995728, -741561803, 1105195401, 425358570

Iteration 64, W =
1633771904, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 24, 1633771904, 983040, -1662491452, 1610613702, 1054730224, 25426944, 1326034916, -488521330, -500405234, 1974893088, 465447438, 824656355, -775676638, -1748277429, -1612118674, 371502742, 962019307, 1644127862, -968364684, 806071431, -519949981, -271404098, 902720945, 1179569674, 275079462, -1869111317, 2025592049, -1490431245, -1500271359, 809604958, -1903267858, 1047881184, -42360992, 51922169, 1293266598, 1497201556, 1084406760, -1865537359, 782543434, 731454045, -156312910, 1962396194, 256089834, 15995728, -741561803, 1105195401, 425358570, -174571706

SHA-256 example values (step 13, then step 14 - 20):

tmp = 1779033703, -1150833019, 1013904242, -1521486534, 1359893119, -1694144372, 528734635, 1541459225

Iteration 1: tmp =
1567222221, 1779033703, -1150833019, 1013904242, -97958878, 1359893119, -1694144372, 528734635

Iteration 2: tmp =
920587069, 1567222221, 1779033703, -1150833019, 961309601, -97958878, 1359893119, -1694144372

Iteration 3: tmp =
109373937, 920587069, 1567222221, 1779033703, -617599971, 961309601, -97958878, 1359893119

Iteration 4: tmp =
351007453, 109373937, 920587069, 1567222221, 1251550470, -617599971, 961309601, -97958878

Iteration 5: tmp =
925844847, 351007453, 109373937, 920587069, 28316548, 1251550470, -617599971, 961309601

Iteration 6: tmp =
-1732995595, 925844847, 351007453, 109373937, 1564207672, 28316548, 1251550470, -617599971

Iteration 7: tmp =
-584050822, -1732995595, 925844847, 351007453, -1288768333, 1564207672, 28316548, 1251550470

Iteration 8: tmp =
1093405581, -584050822, -1732995595, 925844847, -333651519, -1288768333, 1564207672, 28316548

Iteration 9: tmp =
1378357318, 1093405581, -584050822, -1732995595, -200016079, -333651519, -1288768333, 1564207672

Iteration 10: tmp =
-2026955876, 1378357318, 1093405581, -584050822, 398594602, -200016079, -333651519, -1288768333

Iteration 11: tmp =
1677851746, -2026955876, 1378357318, 1093405581, -1127978291, 398594602, -200016079, -333651519

Iteration 12: tmp =
1736103297, 1677851746, -2026955876, 1378357318, -459607487, -1127978291, 398594602, -200016079

Iteration 13: tmp =
-1491468256, 1736103297, 1677851746, -2026955876, 1938991035, -459607487, -1127978291, 398594602

Iteration 14: tmp =
21602884, -1491468256, 1736103297, 1677851746, -526733444, 1938991035, -459607487, -1127978291

Iteration 15: tmp =
1792221982, 21602884, -1491468256, 1736103297, 1920937127, -526733444, 1938991035, -459607487

Iteration 16: tmp =
816479728, 1792221982, 21602884, -1491468256, -1510061704, 1920937127, -526733444, 1938991035

Iteration 17: tmp =
-1210181089, 816479728, 1792221982, 21602884, -204465154, -1510061704, 1920937127, -526733444

Iteration 18: tmp =
1455450966, -1210181089, 816479728, 1792221982, -540177452, -204465154, -1510061704, 1920937127

Iteration 19: tmp =
-407432703, 1455450966, -1210181089, 816479728, 1292878908, -540177452, -204465154, -1510061704

Iteration 20: tmp =
1221141611, -407432703, 1455450966, -1210181089, 1274186142, 1292878908, -540177452, -204465154

Iteration 21: tmp =
235139774, 1221141611, -407432703, 1455450966, -1457043414, 1274186142, 1292878908, -540177452

Iteration 22: tmp =
495184876, 235139774, 1221141611, -407432703, 201765007, -1457043414, 1274186142, 1292878908

Iteration 23: tmp =
-679637464, 495184876, 235139774, 1221141611, 1526797993, 201765007, -1457043414, 1274186142

Iteration 24: tmp =
275104589, -679637464, 495184876, 235139774, -2021031471, 1526797993, 201765007, -1457043414

Iteration 25: tmp =
-966129355, 275104589, -679637464, 495184876, -175413045, -2021031471, 1526797993, 201765007

Iteration 26: tmp =
320008802, -966129355, 275104589, -679637464, -904336099, -175413045, -2021031471, 1526797993

Iteration 27: tmp =
867430204, 320008802, -966129355, 275104589, -1666118212, -904336099, -175413045, -2021031471

Iteration 28: tmp =
1282457349, 867430204, 320008802, -966129355, -1896742784, -1666118212, -904336099, -175413045

Iteration 29: tmp =
1125133851, 1282457349, 867430204, 320008802, -1937973577, -1896742784, -1666118212, -904336099

Iteration 30: tmp =
788740998, 1125133851, 1282457349, 867430204, 507445440, -1937973577, -1896742784, -1666118212

Iteration 31: tmp =
-2104770510, 788740998, 1125133851, 1282457349, 202359331, 507445440, -1937973577, -1896742784

Iteration 32: tmp =
-1498874639, -2104770510, 788740998, 1125133851, 1642095500, 202359331, 507445440, -1937973577

Iteration 33: tmp =
-1874482004, -1498874639, -2104770510, 788740998, -17492684, 1642095500, 202359331, 507445440

Iteration 34: tmp =
-108780555, -1874482004, -1498874639, -2104770510, 1312167812, -17492684, 1642095500, 202359331

Iteration 35: tmp =
-1409732727, -108780555, -1874482004, -1498874639, 213856140, 1312167812, -17492684, 1642095500

Iteration 36: tmp =
764494952, -1409732727, -108780555, -1874482004, 677895976, 213856140, 1312167812, -17492684

Iteration 37: tmp =
541693465, 764494952, -1409732727, -108780555, -1714848330, 677895976, 213856140, 1312167812

Iteration 38: tmp =
1364264462, 541693465, 764494952, -1409732727, 1115234479, -1714848330, 677895976, 213856140

Iteration 39: tmp =
-2103289234, 1364264462, 541693465, 764494952, 755732897, 1115234479, -1714848330, 677895976

Iteration 40: tmp =
-1035864671, -2103289234, 1364264462, 541693465, -186848648, 755732897, 1115234479, -1714848330

Iteration 41: tmp =
-765764850, -1035864671, -2103289234, 1364264462, 1444590219, -186848648, 755732897, 1115234479

Iteration 42: tmp =
763400386, -765764850, -1035864671, -2103289234, -1157514158, 1444590219, -186848648, 755732897

Iteration 43: tmp =
-792099381, 763400386, -765764850, -1035864671, -2114466251, -1157514158, 1444590219, -186848648

Iteration 44: tmp =
-216850192, -792099381, 763400386, -765764850, -401621942, -2114466251, -1157514158, 1444590219

Iteration 45: tmp =
1873674834, -216850192, -792099381, 763400386, -658028235, -401621942, -2114466251, -1157514158

Iteration 46: tmp =
1116552225, 1873674834, -216850192, -792099381, 1259411383, -658028235, -401621942, -2114466251

Iteration 47: tmp =
2027690457, 1116552225, 1873674834, -216850192, 1622974665, 1259411383, -658028235, -401621942

Iteration 48: tmp =
-1117172804, 2027690457, 1116552225, 1873674834, 1700048614, 1622974665, 1259411383, -658028235

Iteration 49: tmp =
-1721218143, -1117172804, 2027690457, 1116552225, -1195906838, 1700048614, 1622974665, 1259411383

Iteration 50: tmp =
211802405, -1721218143, -1117172804, 2027690457, -78878274, -1195906838, 1700048614, 1622974665

Iteration 51: tmp =
-1911910619, 211802405, -1721218143, -1117172804, 1947358932, -78878274, -1195906838, 1700048614

Iteration 52: tmp =
721125858, -1911910619, 211802405, -1721218143, 734510292, 1947358932, -78878274, -1195906838

Iteration 53: tmp =
-960963640, 721125858, -1911910619, 211802405, -438477003, 734510292, 1947358932, -78878274

Iteration 54: tmp =
-2021631552, -960963640, 721125858, -1911910619, -119634217, -438477003, 734510292, 1947358932

Iteration 55: tmp =
1103893989, -2021631552, -960963640, 721125858, 1520083678, -119634217, -438477003, 734510292

Iteration 56: tmp =
-1007258133, 1103893989, -2021631552, -960963640, 1482358767, 1520083678, -119634217, -438477003

Iteration 57: tmp =
32012902, -1007258133, 1103893989, -2021631552, 2120798855, 1482358767, 1520083678, -119634217

Iteration 58: tmp =
-312093446, 32012902, -1007258133, 1103893989, 1079389942, 2120798855, 1482358767, 1520083678

Iteration 59: tmp =
-1047518888, -312093446, 32012902, -1007258133, -1248211794, 1079389942, 2120798855, 1482358767

Iteration 60: tmp =
1349816202, -1047518888, -312093446, 32012902, -664589014, -1248211794, 1079389942, 2120798855

Iteration 61: tmp =
-1365528494, 1349816202, -1047518888, -312093446, -1264706601, -664589014, -1248211794, 1079389942

Iteration 62: tmp =
725012183, -1365528494, 1349816202, -1047518888, 270086148, -1264706601, -664589014, -1248211794

Iteration 63: tmp =
340307500, 725012183, -1365528494, 1349816202, 1146259971, 270086148, -1264706601, -664589014

Iteration 64: tmp =
774545670, 340307500, 725012183, -1365528494, -105232229, 1146259971, 270086148, -1264706601

SHA-256 example values (H values after step 22):
H = -1741387923, -810525519, 1738916425, 1407952268, 1254660890, -547884401, 798820783, 276752624

H (as a one hex string) =
0x9834876DCFB05CB167A5C24953EBA58C4AC89B1ADF57F28F2F9D09AF107EE8F0

And this is a final hash.

Exercise 1:

Write a program which will produce SHA-256 hash. Check your hashing function for three messages below:

- message = „”
- message = „aaa”
- message = „aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa aaaaa”

Compare your hashes with an online hash calculator.