Course: "Fundamentals of Cryptography".
Laboratory work No. 9. Hash functions

The task 1:

Calculate hash functions for 5 different combinations of **n** and **m** for each of the following tasks as

1.1. h (n) = n mod m

1.2. h (n) = (h (n1) + h (n2) + h (n3) + h (n4) + h (n5) + h (n6)) mod m

The values **n, n1-n6** and **m** are chosen from the tables of the initial IP permutation and the final IP permutation arbitrarily.

Combinations must be calculated for cases when:

A) even **n** and odd **m**;

C) odd **n** and even **m**.

Write a program that implements the implementation of points 1 and 2.

Record the results in the form of a table.

Task 2:

2.1. Suggest your own algorithm for fast transformation of a string of arbitrary length into a hash code.

Task 3:

3.1. Implement Schnopp's digital signature algorithm for arbitrary values of **p** and **q**.

Schnorr's electronic signature algorithm.

At the first stage, the parameters of the algorithm are selected:

p is a simple number, for real problems its length should be 160 bits < p < 256 bits;

q is a prime number chosen so that it is a divisor of the number p-1);

g is chosen according to a special procedure from Z_p, where Z_p is the class of residues modulo **p**;

H – hash function;

x is a random number from the interval [1, q-1]; **x** – private key of the scheme;

y – public key of the scheme;

Y=g-x.

After the parameters are selected, the algorithm operations are performed:

1. Participant A chooses a random number **k** and calculates **r=g*k(mod p)**.

(for this example the number **g** is chosen arbitrarily)

2. Participant A creates a signature, for this purpose **e** and s are calculated according to the formulas

$$e = H(r,A); \quad s = k + xe.$$

3. Signature (e,s) and text M are sent to participant B.

4. Participant B verifies the signature by calculating the values of r′ and e′,

$$r' = g*s*y*e \bmod p;$$

$$e' = h(r',m).$$

5. The values of e and e′ are compared.

The task 4:

2.1. Write a program that implements items 1,2 and 4,5.