1. Cryptography

A. Most modern digital computers and related software include a linear congruential pseudo-random number generator of the recursive form

(2.1) $X_{n+1} = [\ a\ .\ X_n + b\ ]\ (\mathrm{mod}\ c)$

where *a* and *c* are positive integers and *b* is a nonnegative integer. For an integer initial value or seed X_c, the algorithm (2.1) generates a sequence taking integer values from 0 to c -1, the remainders when the $aX_n + b$ are divided by c.

Generate a sequence of 10 pseudo-random numbers by the linear congruential generator (2.1) with a = 1229, b = 1 and c = 2048.

Mod - Remainder after division (modulo operation)
23 mod 5 = 3. 23 / 5 = 4 and rest is 3
12 mod 8 = 4
65 mod 9 = 2
4 mod 2 = 0

2. RSA (cryptosystem), **https://en.wikipedia.org/wiki/RSA_(cryptosystem)**
Example

Here is an example of RSA encryption and decryption. The parameters used here are artificially small,

1. Choose two distinct prime numbers, such as
   $p = 61$ and $q = 53$
2. Compute *n* = *pq* giving
   $n = 61 \times 53 = 3233$
3. Compute the Carmichael's totient function of the product as *λ*(*n*) = lcm(*p* − 1, *q* − 1) giving
   $\lambda(3233) = \mathrm{lcm}(60, 52) = 780$
4. Choose any number 1 < *e* < 780 that is coprime to 780. Choosing a prime number for *e* leaves us only to check that *e* is not a divisor of 780.
   Let $e = 17$
5. Compute *d*, the modular multiplicative inverse of *e* (mod *λ*(*n*)) yielding,
   $d = 413$
   Worked example for the modular multiplicative inverse:
   $d \times e \bmod \lambda(n) = 1$
   $413 \times 17 \bmod 780 = 1$

The **public key** is (*n* = 3233, *e* = 17). For a padded plaintext message *m*, the encryption function is

$$c(m) = m^{17} \bmod 3233$$

The **private key** is (*n* = 3233, *d* = 413). For an encrypted ciphertext *c*, the decryption function is

$$m(c) = c^{413} \bmod 3233$$

For instance, in order to encrypt *m* = 65, we calculate

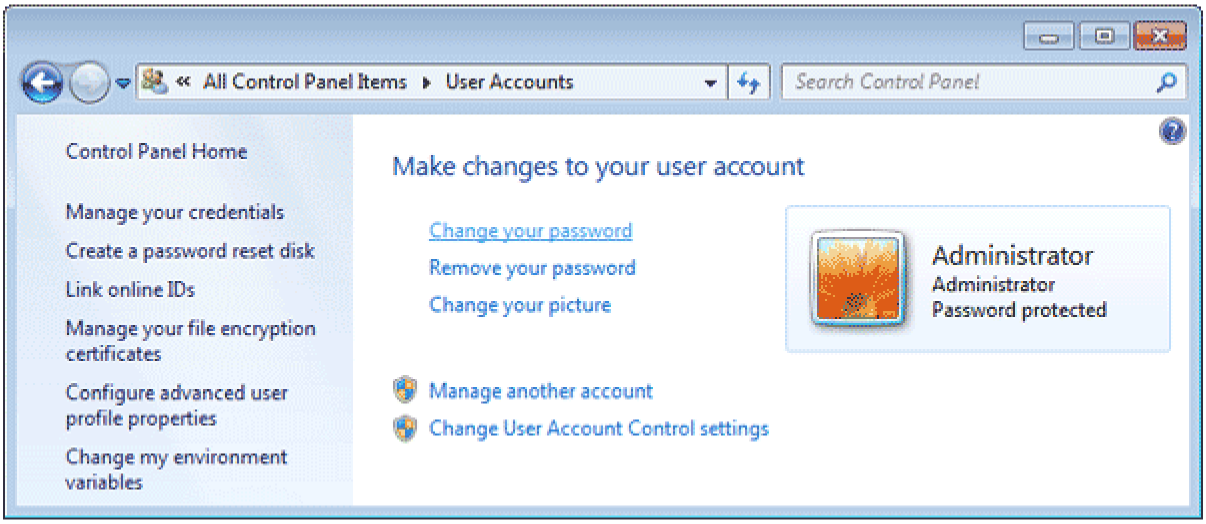
$$c = 65^{17} \bmod 3233 = 2790$$

To decrypt *c* = 2790, we calculate

$$m = 2790^{413} \bmod 3233 = 65$$

New prime numbers p=5, q=11. Encrypt m=65
There is any mistake ☹: n=55; e=7; d=3; lambda=20; c=10.

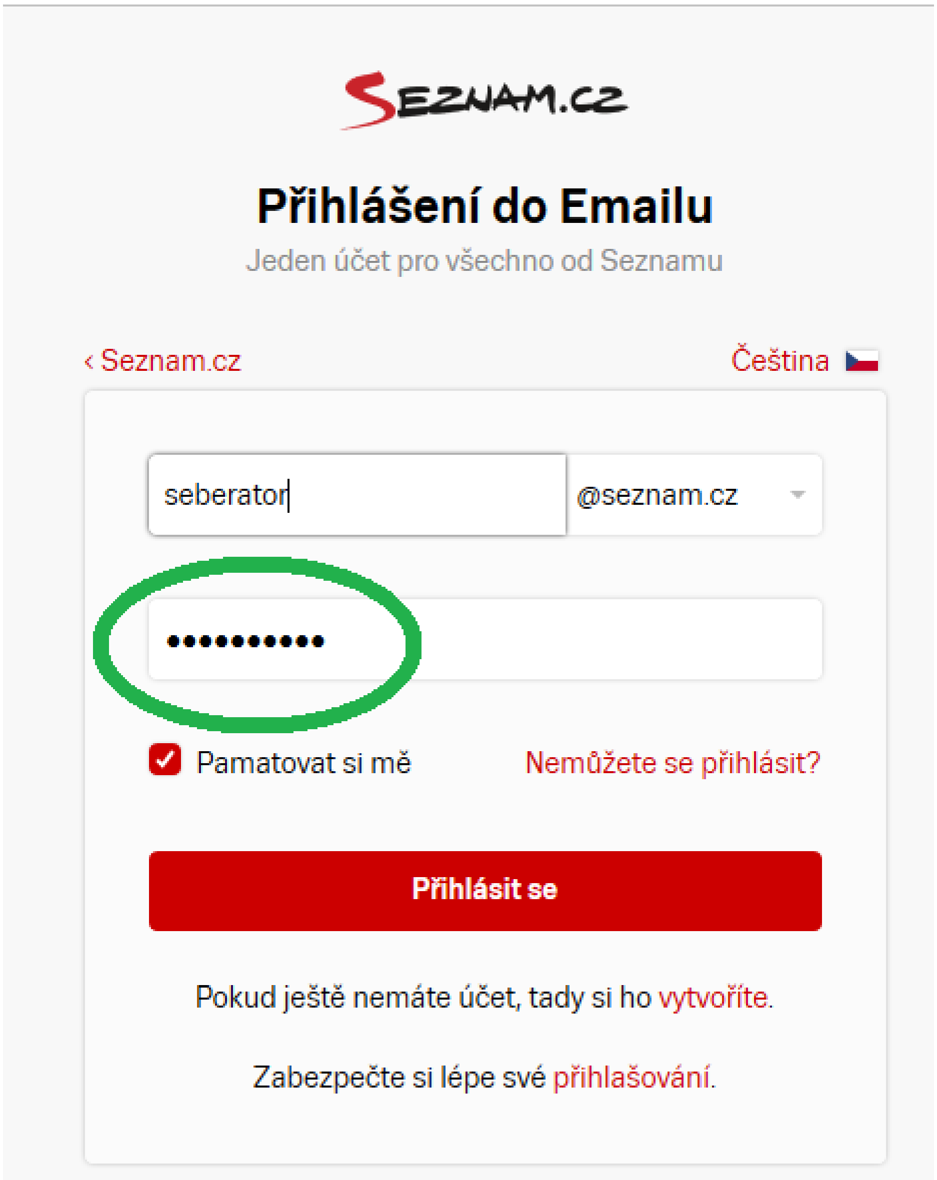
3. Password in Windows users



User lost password to windows account. Do boot flash disk and find password ☺.
Use only free sw…, for example http://ophcrack.sourceforge.net/

4. Password hidden behind asterisks

How To Reveal The Password Hidden Behind Asterisks



1. **Super Team**

3 + 4

2. **Supreme**

2 + 4

3. **Google team**

1 + 4