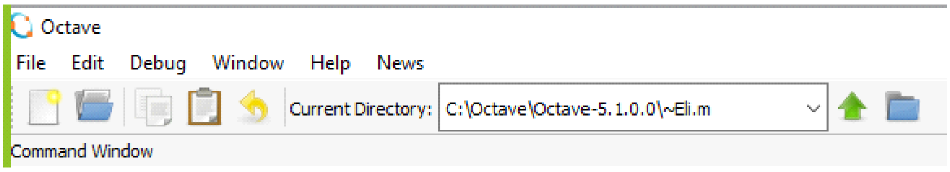
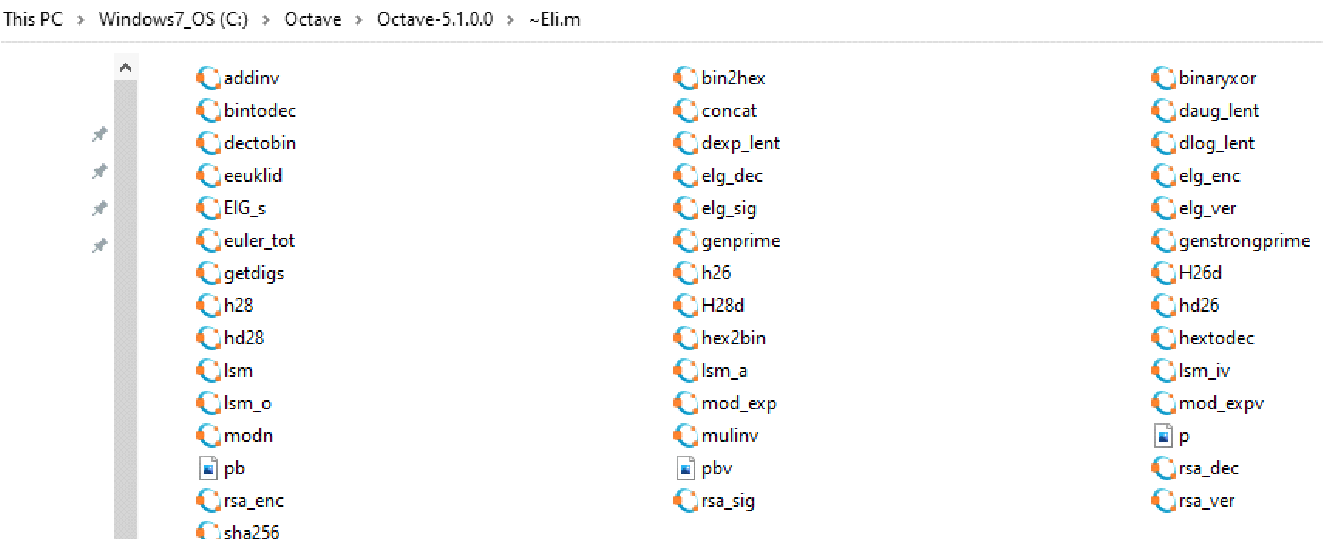
http://crypto.fmf.ktu.lt/xdownload/

- octave-5.1.0-w64-installer.exe
- octave-5.1.0.pdf
- Octave_Stud_2020.05.7z



C:\Octave\Octave-5.1.0.0\~Eli.m



Decimal	Binary	Hexadecimal
0	0_b	0_h
0+1=1	$0+1=1_b$	$0+1=1_h$
1+1=2	$1+1=10_b$	$1+1=2_h$
2+1=3	$10+1=11_b$	$2+1=3_h$
3+1=4	$11+1=100_b$	$3+1=4_h$
4+1=5	$100+1=101_b$	$4+1=5_h$
5+1=6	$101+1=110_b$	$5+1=6_h$
6+1=7	$110+1=111_b$	$6+1=7_h$
7+1=8	$111+1=1000_b$	$7+1=8_h$
8+1=9	$1000+1=1001_b$	$8+1=9_h$
9+1=10	$1001+1=1010_b$	$9+1=A_h$
10+1=11	$1010+1=1011_b$	$A+1=B_h$
11+1=12	$1011+1=1100_b$	$B+1=C_h$
12+1=13	$1100+1=1101_b$	$C+1=D_h$
13+1=14	$1101+1=1110_b$	$D+1=E_h$
14+1=15	$1110+1=1111_b$	$E+1=F_h$
15+1=16	$1111+1=10000_b$	$F+1=10_h$
16+1=17	10000+1=10001	10+1=11

positional system of numbers

Decimal numbers has a base = 10

$17 = 1\cdot 10^1 + 7\cdot 10^0 = 10+7$

least significant digit

most significant digit

Binary numbers has a base = 2

Hexadecimal numbers has a base = 16

```
>> d=17
d =  17
>> db=dec2bin(d)
db = 10001
>> dh=dec2hex(d)
dh = 11
```

```
>> d1=bin2dec(db)
d1 =  17
>> d2=hex2dec(dh)
d2 =  17
>> db1=hex2bin(dh)
db1 = 10001
>> dh1=bin2hex(db1)
dh1 = 11
```

14+1=15	$1110+1=1111_b$	$E+1=F_h$
15+1=16	$1111+1=10000_b$	$F+1=10_h$
16+1=17	$10000+1=10001_b$	$10+1=11_h$

```
dh = 11
db1 = 10001
>> dh1=bin2hex(db1)
dh1 = 11
```

$|15| = 4$ bits ; $15 \equiv 1111 \equiv 2^4 - 1 = 16 - 1 = 15$

Let we want to contruct max number of q bits, then it is equal to $2^q - 1$

Let $q = 10 \rightarrow 2^{10} - 1 = 1024 - 1 = 1023$.

$$\begin{array}{r} 1111111111 \\ +\qquad\qquad 1 \\ \hline 10000000000 \end{array}$$

```
>> q=10
q =  10
>> 2^q-1
ans =  1023
>> z=ans
z =  1023
>> dec2bin(z)
ans = 1111111111
```

$$17 = \overset{4}{1}\overset{3}{0}\overset{2}{0}\overset{1}{0}\overset{0}{1}_b = 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 2^4 + 2^0 = 16 + 1 = 17$$

$$17 \equiv \overset{1}{1}\overset{0}{1}_h = 1 \cdot 16^1 + 1 \cdot 16^0 = 16 + 1 = 17$$

$3 \cdot 11 = 33$

Modular arithmetics

Let we fix some integer n, and have some integer Z. Then Z can expressed in unique form by

$$Z = k \cdot n + r$$

$$Z \bmod n = r$$

$$\begin{array}{r|l} k \cdot n + r & n \\ \cline{2-2} -\; k \cdot n & k \\ \cline{1-1} r & \end{array}$$

Let $n = 15$, $Z = 33 \rightarrow Z = 2 \cdot 15 + 3$

$$\begin{array}{r|l} 33 & 15 \\ \cline{2-2} -\;30 & 2 \\ \cline{1-1} 3 & \end{array}$$

n - module in modular arithmetics.

$33 \bmod 15 = 3$

```
>> z=33
z =  33
>> n=15
n =  15
>> zmn=mod(z,n)
zmn = 3
```

Random number generation

We will deal with the numbers of 28 bi length in Octave.

We generate random number do not exceeding 28 bit length

$|rb| = 24 < 28$ bits

```
>> r=randi(2^28-1)
r =  16332653
>> rb=dec2bin(r)
rb = 111110010011011101101101
```

```
>> zmax=2^28-1
zmax =  268 435 455
>> zb=dec2bin(zmax)
zb = 1111111111111111111111111111
```