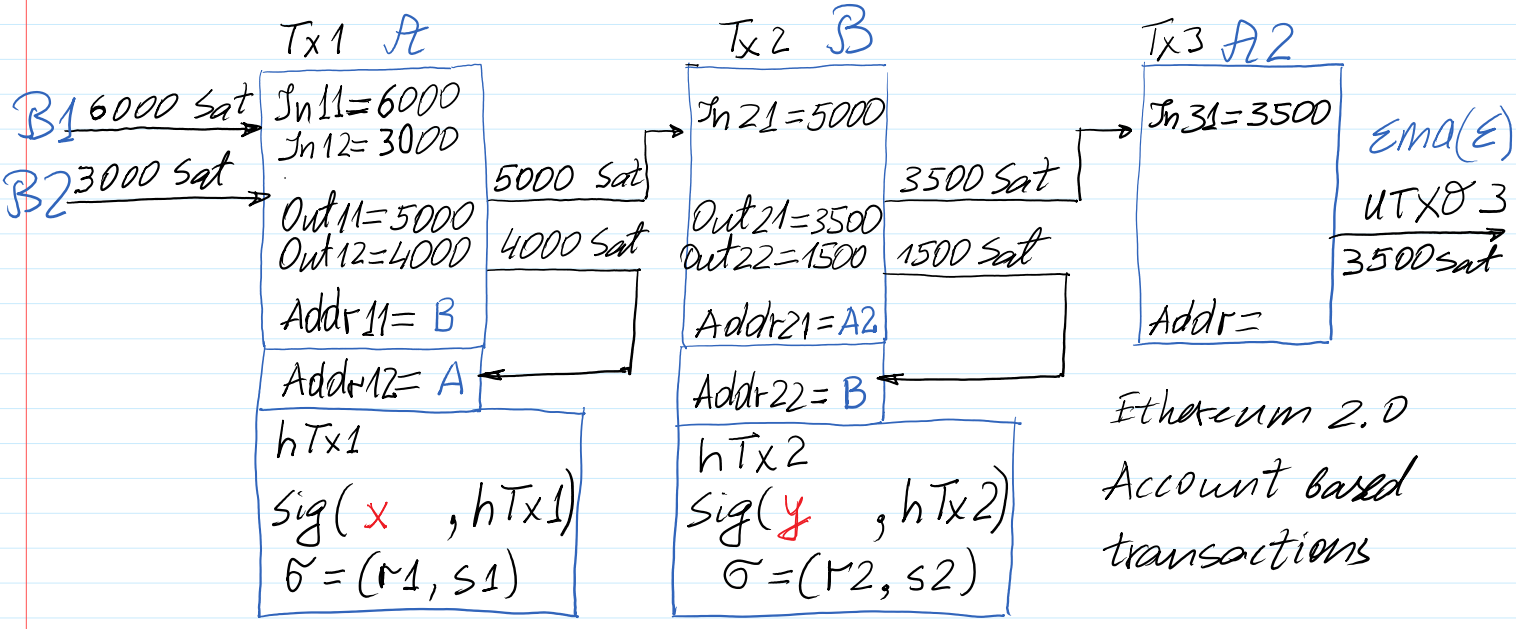
Kursinių darbų gynimas: Gruodžio 13, 20 d., 17:30 per Zoom.

Public Parameters **PP = (p, g); p = 268 435 019; g = 2;**
p=int64(268435019)
p = 268435019
>> g=2;

Unspent Transactions Output – UTxO



'Tx1 : In11 = 6000 || In12 = 3000 || Out11 = 5000 || Out12 = 4000 || Rec1 = B || Rec2 = A'

$hTx1 = h28(\downarrow)$

$Sign(PrK_A = x, hTx1) = \curvearrowleft S_1 = \sigma_1 = (r_1, s_1)$ % Schnorr Sig.

'Tx2 : In21 = 5000 || Out21 = 3500 || Out22 = 1500 || Rec1 = A2 || Rec2 = B'

$hTx2 = h28('\downarrow')$

$sign(PrK_B = y, h2Sig) = \curvearrowleft S_2 = \sigma_2 = (r_2, s_2)$ % Schnorr Sig.

>> x=int64(randi(2^27)) >> y=int64(randi(2^27))

>> x=int64(randi(2^27))
x = 100497451
>> a=mod_exp(g,x,p)
a = 91968695

>> y=int64(randi(2^27))
y = 25824381
>> b=mod_exp(g,y,p)
b = 195335035

Schnorr Signature ← **h** - represents message to be signed
Signature is denoted by **Ϭ=(r, s)**

u=randi(p-1)
r=g^u mod p
h=hd28(M||r) % **h** is a decimal number
>> h=hd28(concat(M,r))
s=u+xh mod (p-1)

Verification of **Ϭ=S=(r,s)** for **h=hd28(M||r)**

$$g^s = ra^h \bmod p$$

Schnorr Signature ← **h** - represents h-value to be signed in decimal format

Transaction Tx1 signing: **h1=hd28(Tx1) =** 184316888
hTx1d=hd28(Tx1)
u1=randi(p-1)
r1=mod_exp(g,u1,p)
h1=hd28(hTx1d||r1) % **h1** is a decimal number
s1=u1+x(h1) mod (p-1)

Signature for Tx1 computation **$Ϭ_1$=S1=(r1, s1)**

Signature fot Tx2 computation **$Ϭ_2$=S2=(r2, s2)**

>> u1=randi(p-1)
u1 = 96927099
>> r1=mod_exp(g,u1,p)
r1 = 240239134
>> h1=hd28(hTx1d||r1)
h1 = 126174618
>> xh1=mod(x*h1,p-1)
xh1 = 319146
>> s1=mod(u1+xh1,p-1)
s1 = 97246245

$$g^s = ra^h \bmod p$$

$$g^{s1} = (r1)a^{h1} \bmod p$$

```
>> g_s1=mod_exp(g,s1,p)
g_s1 = 168542564
>> a
a = 91968695
>> a_h1=mod_exp(a,h1,p)
a_h1 = 128485502
>> r1
r1 = 240239134
>> r1a_h1=mod(r1*a_h1,p)
r1a_h1 = 168542564
```

```
>> hTx1d=hex2dec('AFC73D8')
hTx1d =  184316888
```

%%% Signature **Sig1** Creation on **hTx1d**
Transaction Tx1 signing: **M=hTx1d**

```
u1=randi(p-1)
r1=mod_exp(g,u1,p)
h1=hd28('hTx1d||r')    % h1 is a decimal number
xh1=mod(x*h1,p-1)
s1=mod(u1+xh1,p-1)
```

Sig1=(r1, s1)

Signature **Sig1** Verification

%%% Signature verification

```
>> hTx2d=hex2dec('43BC2D0')
hTx2d =  71025360
```

%%% Signature **Sig2** Creation on **hTx2d**
Transaction Tx2 signing: **M=hTx2d**

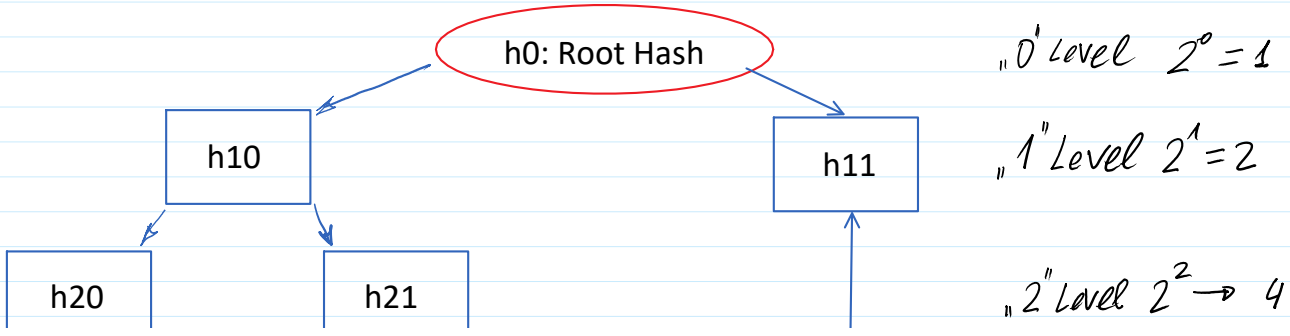
```
u2=randi(p-1)
r2=mod_exp(g,u1,p)
h2=hd28('hTx2d||r')    % h1 is a decimal number
xh2=mod(y*h2,p-1)
s1=mod(u1+xh2,p-1)
```

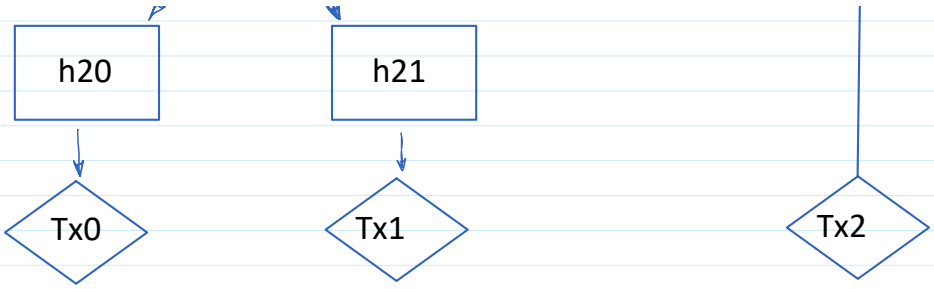
Sig2=(r2, s2)

Signature **Sig2** Verification

%%% Signature verification

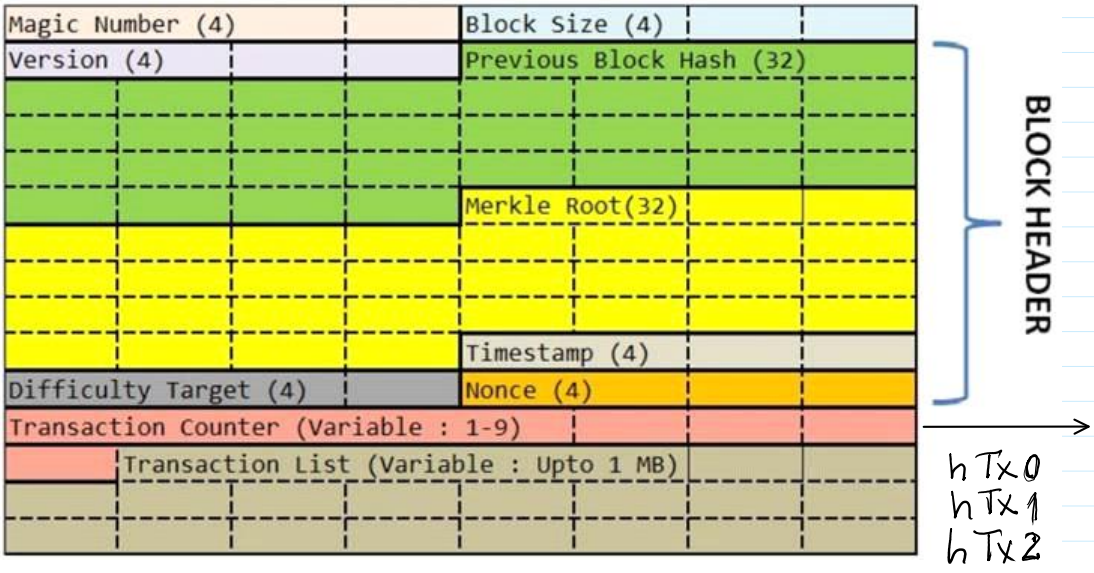
Transaction template:
TxN = 'TxN:InN1=…||InN2=…||OutN1=…||OutN2=…||RecN1=…||RecN2=…'
Transactions:
Tx0 = 'Tx0:In01=6000||In02=3000||Out01=5000||Out02=4000||Rec1=B||Rec2=A'
Tx1='Tx1:In11=5000||Out11=3500||Out12=1500||Rec1=A2||Rec2=B'
Tx2='Tx2:In21=3500||Out21=3500||Out22=0||Rec1=E||Rec2=0'





„2" Level $2^2 \rightarrow 4$

>> h20=h28(Tx0)
h20 = 8046704
>> h21=h28(Tx1)
h21 = FB4B763
>> h11=h28(Tx2)
h11 = 3619B27
>>
>> h10=h28('8046704||FB4B763')
h10 = D646CB9
>> RH=h28('D646CB9||3619B27')
RH = EF0B3B4



SHA-256

PrBlH = 0CAF06F
RH = EF0B3B4

Difficulty Target (DT): we define as 1 leading zero hex digit in block h-value.

Block data:
Bl1='Bl1:PrBlh=0CAF06F||RH=EF0B3B4||hTx0=8046704||hTx1=FB4B763||hTx2=3619B27||nonce=1000'
Bl1=h28(Bl1)
Bl1 = 676F37F

>> Bl1='Bl1:PrBlh=0CAF06F||RH=EF0B3B4||hTx0=8046704||hTx1=FB4B763||hTx2=3619
B27||nonce=1001'
Bl1 = Bl1:PrBlh=0CAF06F||RH=EF0B3B4||hTx0=8046704||hTx1=FB4B763||hTx2=3619B27
||nonce=1001
>> hBl1=h28(Bl1)
hBl1 = DDDDDB6

Mining: Difficulty Target (DT).

Bl2:PrBlh=0CAF06F||RH=EF0B3B4||hTx0=8046704||hTx1=FB4B763||hTx2=3619B27||nonce=2011
nonce: 2011
Bl2: 0516A3C

6 hex numb. x 4 bits = 24 bits $\longrightarrow 2^{24}$

4 bits $\longrightarrow 2^4 = 16$

nonce: 2235
Bl2: 00B7B95

nonce: 173147
Bl2: 0000428

nonce: 82 719 324
Bl2: 0000000

The number of possible h-values of 28 bits length >> 2^28 ans = 268 435 456

The number of adequate h-values: 2^{24}

$$Pr\{to\ Mine\} = \frac{2^{24}}{2^{28}} = \frac{1}{2^4} = \frac{1}{16}$$

The number of adequate h-values: 2^{20}

$$Pr\{to\ Mine\} = \frac{2^{20}}{2^{28}} = \frac{1}{2^8} = \frac{1}{256}$$

$: 2^{12}$

$$Pr\{to\ Mine\} = \frac{2^{12}}{2^{28}} = \frac{1}{2^{16}} = \frac{1}{65536}$$ >> 2^16 ans = 65536

$: 1$

$$Pr\{to\ Mine\} = \frac{1}{\quad} = \frac{1}{\quad}$$

$$\Pr\{\text{to Mine}\} = \frac{1}{2^{28}} = \frac{1}{268\,435\,456} \quad : 1$$

>> 2^28 ans = 268 435 456

```
>> Bl1='Bl1:PrBlh=0CAF06F||RH=EF0B3B4||hTx0=8046704||hTx1=FB4B763||hTx2=3619
B27||nonce=1007'
Bl1 = Bl1:PrBlh=0CAF06F||RH=EF0B3B4||hTx0=8046704||hTx1=FB4B763||hTx2=3619B27
||nonce=1007
>> h=sha256(Bl1)
h = 85EDE1076C7B246DEC 0EB6B3273919E3114937AAC9BBE461A77574840A9BC018
```

"0"

184 bits

Bitcoin mining: h-value has 256 bits SHA 256

64 hex digits

Difficulty Target: to get 18 leading hex digits equal to „0".

72 leading binary digits

Total number of h-values: $2^{256} \sim 10^{80}$

Adequate number of h-values: $256 - 72 = 184$ bits $\Rightarrow 2^{184}$

$$\Pr\{\text{Mining}\} = \frac{2^{184}}{2^{256}} = \frac{1}{2^{72}} \; ; \quad 2^{72} =$$

>>> pow(2,72) = 4 722 366 482 869 645 213 696 ~~ 10^21

>> 2^72 ans = 4.7224e+21

>> int64(2^72) ans = 9223372036854775807