

Template for mining is presented in:

https://docs.google.com/spreadsheets/d/1Sb291NUz8lgmfDlx_QndCdqsJ9CWAhUp/edit?usp=sharing&oid=111502255533491874828&rtpof=true&sd=true

Tx1:In11=6000|In12=3000|Out11=5000|Out12=4000|Rec1=2|Rec2=1

Address_N	PuK_N	InN11	InN12	OutN11	OutN12	RecN1	RecN2
		6000	3000	5000	4000	2	1

Tx_1='Tx1:In11=6000|In12=3000|Out11=5000|Out12=4000|Rec1=2|Rec2=1'

>> Tx_1='Tx1:In11=6000|In12=3000|Out11=5000|Out12=4000|Rec1=2|Rec2=1'

Tx_1 = Tx1:In11=6000|In12=3000|Out11=5000|Out12=4000|Rec1=2|Rec2=1

>> hTx_1=h28(Tx_1)

hTx_1 = 3A40657

>> Tx_2='Tx2:In11=5000|In12=1000|Out11=2000|Out12=4000|Rec1=3|Rec2=1'

Tx_2 = Tx2:In11=5000|In12=1000|Out11=2000|Out12=4000|Rec1=3|Rec2=1

>> Tx_3='Tx3:In11=2000|In12=4000|Out11=6000|Out12=0|Rec1=4|Rec2=2'

Tx_3 = Tx3:In11=2000|In12=4000|Out11=6000|Out12=0|Rec1=4|Rec2=2

>> hTx_2=h28(Tx_2)

hTx_2 = AA19763

>> hTx_3=h28(Tx_3)

hTx_3 = F098BFD

>>

>> cc12=concat(hTx_1,hTx_2)

cc12 = 3A40657AA19763

>>>> hTx_12=h28(cc12)

hTx_12 = 90BEF05

>> cc12_3=concat(hTx_12,hTx_3)

cc12_3 = 90BEF05F098BFD

>> hTx_12_3=h28(cc12_3)

hTx_12_3 = 4B6968F

Root hash = 4B6968F

>> Bl_23456789='hPrBl=0CAF06F|Root h=4B6968F|nonce=1000'

Bl_23456789 = hPrBl=0CAF06F|Root h=4B6968F|nonce=1000

>>

>> Bl_23456789='hPrBl=0CAF06F|Root h=4B6968F|nonce=1000'

Bl_23456789 = hPrBl=0CAF06F|Root h=4B6968F|nonce=1000

>>

>> Mining=h28('hPrBl=0CAF06F|Root h=4B6968F|nonce=1000')

Mining = A3AEFF1

>> Mining=h28('hPrBl=0CAF06F|Root h=4B6968F|nonce=1001')

Mining = 9215560

>> Mining=h28('hPrBl=0CAF06F|Root h=4B6968F|nonce=1002')

```

Mining = 9EAA330
>> Mining=h28('hPrBl=0CAF06F | |Root h=4B6968F | |nonce=1003')
Mining = 53F64D3
>> Mining=h28('hPrBl=0CAF06F | |Root h=4B6968F | |nonce=1004')
Mining = A85EA95
>> Mining=h28('hPrBl=0CAF06F | |Root h=4B6968F | |nonce=1005')
Mining = CD1AE9D
>> Mining=h28('hPrBl=0CAF06F | |Root h=4B6968F | |nonce=1006')
Mining = E403FA8
>> Mining=h28('hPrBl=0CAF06F | |Root h=4B6968F | |nonce=1007')
Mining = 097E03D

```

Winning miner declares the following data to Net
Mining=h28('hPrBl=0CAF06F | |Root h=4B6968F | |nonce=1007')
Mining = 097E03D

<https://medium.com/codechain/modified-merkle-patricia-trie-how-ethereum-saves-a-state-e6d7555078dd>

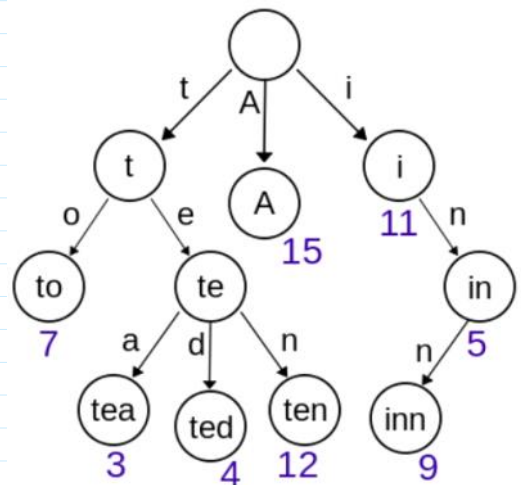
Modified Merkle Patricia Trie (a.k.a MPT) as the method to save Ethereum state.

Basically, MPT is a combination of Patricia trie and Merkle tree, with few additional optimizations that fit the characteristics of Ethereum. Thus, an understanding of the Patricia trie and Merkle tree should precede the understanding of MPT.

Patricia Trie

Patricia trie is a data structure which is also called Prefix tree, radix tree or trie. Trie uses a key as a path so the nodes that share the same prefix can also share the same path. This structure is **fastest** at finding common prefixes, simple to implement, and requires small memory. Thereby, it is commonly used for implementing routing tables, systems that are used in low specification machines like the router.

Trie ← Retrieve



Example of Patricia Trie

Till this place

TxN:InN1=... InN2=... OutN1=... OutN2=... RecN1=... RecN2=...	hTx_N	N1	N2	N3	H(TxN1) hTx_N1	H(TxN2) hTx_N2	H(TxN3) hTx_N3
Tx1:In11=6000 In12=3000 Out11=5000 Out12=4000 Rec1=B Rec2=A	AFC73D8	1	2	3	AFC73D8	13251F8	5B5412B
Tx2:In21=5000 Out21=3500 Out22=1500 Rec1=...	432E1E9						

Tx1:In11=6000 In12=3000 Out11=5000 Out12=4000 Rec1=B Rec2=A	AFC73D8	1	2	3	AFC73D8	13251F8	5B5412B
Tx2:In21=5000 Out21=3500 Out22=1500 Rec1=A2 Rec2=B	13251F8	<i>changed</i>					
Tx3:In31=3500 Out31=3500 Out32=0 Rec1=E Rec2=A2	5B5412B						

hPrBl	hRoot	Bl_N:hPrBl=0CAF06F hRoot=2CC219F hTx_N1=AFC73D8 hTx_N2=13251F8 hTx_N3=5B5412B Nonce=1000	hBl_N	Nonce	hBl_N_Mined
0CAF06F	2CC219F	Bl_1:hPrBl=0CAF06F hRoot=2CC219F hTx_1=AFC73D8 hTx_2=13251F8 hTx_3=5B5412B Nonce=1000		1021	06F61B0

Bl_1:hPrBl=0CAF06F | |hRoot=2CC219F | |hTx_1=AFC73D8 | |hTx_2=13251F8 | |hTx_3=5B5412B | |Nonce=1000

```
>> Bl_1Mng='Bl_1:hPrBl=0CAF06F | |hRoot=2CC219F | |hTx_1=AFC73D8 | |hTx_2=13251F8 | |hTx_3=5B5412B | |Nonce=1000'
Bl_1Mng = Bl_1:hPrBl=0CAF06F | |hRoot=2CC219F | |hTx_1=AFC73D8 | |hTx_2=13251F8 | |hTx_3=5B5412B | |Nonce=1000
```

For mining h - value of Bl_1Mng must be computed according to Difficulty Target (DT)

```
>> hBl_1Mining=h28(Bl_1Mining)
hBl_1Mining = 2520EB3
>> hBl_1Mining=h28(Bl_1Mining)
hBl_1Mining = D4FB37A
```

```
>> hBl_1Mng=h28('Bl_1:hPrBl=0CAF06F | |hRoot=2CC219F | |hTx_1=AFC73D8 | |hTx_2=13251F8 | |hTx_3=5B5412B | |Nonce=1000')
hBl_1Mng = 2520EB3
```

```
>> hBl_1Mng=h28('Bl_1:hPrBl=0CAF06F | |hRoot=2CC219F | |hTx_1=AFC73D8 | |hTx_2=13251F8 | |hTx_3=5B5412B | |Nonce=1001')
hBl_1Mng = D4FB37A
```

```
>> hBl_1Mng=h28('Bl_1:hPrBl=0CAF06F | |hRoot=2CC219F | |hTx_1=AFC73D8 | |hTx_2=13251F8 | |hTx_3=5B5412B | |Nonce=1002')
hBl_1Mng = 6EC93FC
```

```
>> hBl_1Mng=h28('Bl_1:hPrBl=0CAF06F | |hRoot=2CC219F | |hTx_1=AFC73D8 | |hTx_2=13251F8 | |hTx_3=5B5412B | |Nonce=1003')
hBl_1Mng = 98A6656
```

After 22 trials when Nonce=1021 the block is mined:

```
>> hBl_1Mng=h28('Bl_1:hPrBl=OCAF06F||hRoot=2CC219F||hTx_1=AFC73D8||hTx_2=13251F8||hTx_3=5B5412B||Nonce=1021')
```

hBl_1Mng = 06F61B0

After 22 trials when Nonce=1021 the block is mined.