



hd28.m - computing 28 bit length h-value in decimal form

h28.m - computing 28 bit length h-value in hexadecimal form

sha256.m - computing 256 bit length h-value in hexadecimal form



Blockchain

Simplified and symbolic block data: just for illustration of mining process in e.g. **Bitcoin**
Using consensus mechanism **Proof-of-Work - PoW**.

```
>> h=sha256('PrevHash || MerkleRoot || Nonce=737327631')
ans = F4AE534CD226FAF7998C8424B348E020BA80639A687E93A0B8C5130EDC51E6DE
>> h28('RootHash PrevHash 737327631')
ans = C51E6DE
>> hd28('RootHash PrevHash 737327631')
ans = 206694110
>> dec2bin(ans)
```

```
ans = 1100010100011110011011011110
```

```
>> dec2hex(206694110)
```

```
ans = C51E6DE
```

```
>> h=hd28('Hello Bob')
```

```
h = 198770750
```

```
>> dec2bin(h)
```

```
ans = 1011110110010000000000111110
```

```
>> h=h28('Hello Bob')
```

```
h = BD9003E
```

```
>> hex2bin(h)
```

```
ans = 1011110110010000000000111110
```

Example is changed from audio/video record.

Nonce=111 000 000 for e.g. Consensus Mechanism in Bitcoin for PoW mining.

Difficulty Target is 18 leading zeroes of hexadecimal digits in h-value.

In this example we are "mining" a block trying to find h-value with a single leading hexadecimal zero digit.

```
>> h=sha256('PreviousBlockHash| |MerkleRoot| |Nonce=111000000')
```

```
h = 2EC57A9F0FAE29ACA611021071C07F26DDC200AC8FA334A1F409E31A59549453
```

```
>> h=sha256('PreviousBlockHash| |MerkleRoot| |Nonce=111000001')
```

```
h = F667486476414C260C32AE3CA41A8F9DA9D97DDC3AA2CB20E27C8D5444BB43A0
```

```
>> h=sha256('PreviousBlockHash| |MerkleRoot| |Nonce=111000002')
```

```
h = 52AED6B6DAA67BB70B69A545FEB3201A1B34C3636FC54E8B6F4EBC5EA3E30ECF
```

```
>> h=sha256('PreviousBlockHash| |MerkleRoot| |Nonce=111000003')
```

```
h = B477F42D143558CADF5EA9D96B931E21C6BA4FD67DDB8BE477D211E7BB46067F
```

Continue with other Nonce values.

Repeat these computations with your own data and compute h-value with a single leading hexadecimal zero digit.