

HMAC Documentation

`binstring(self, g, p):`

Input variables: `x(int)`

- Takes a number `x` and outputs its corresponding binary representation in the form of a
- string
- It's a helper function

`genKey(self,n):`

Input variables: `n(int)`

- Generates a random binary string of length `n`

`getint(self,s):`

Input variables: `s(str)`

- Returns the integer value of a string containing the binary representation

`setStrLen(self, s, n):`

Input variables: `s(str)`, `n(int)`

- Sets a binary string to given length `n` and returns it

`getxor(self, s1, s2):`

Input variables: `s1(str)`, `s2(str)`

- Performs xor between two binary strings and returns the outcome in binary string format

`hmac(self, hash, m, k):`

Input variables: `hash(hash obj)`, `m(str)`, `k(str)`

- Implements hmac using the hash object
- Returns the hmac of `m(=t)`