**SHA-512:**

The code is an implementation of the SHA-512 (Secure Hash Algorithm 512-bit) hashing algorithm in Python. SHA-512 is a cryptographic hash function that takes an input message and produces a fixed-size (512-bit) hash value, which is typically represented as a hexadecimal string. Here's an explanation of the functions and classes used in the code:

**Constants K and HASH\_VALUE:**

K: This is a list of 64 constant 64-bit values used in the SHA-512 compression function.

HASH\_VALUE: It's a list of 8 initial hash values, which are continuously updated during the compression process to represent the current state of the hash.

**SHA-512 Functions:**

Ch(e, f, g): A function that implements the Ch operation in SHA-512 compression, generating a 64-bit value from the inputs e, f, and g.

Maj(a, b, c): This function performs the Maj operation, generating a 64-bit value based on inputs a, b, and c.

rotr(x, n): It's a right-rotate function that shifts the bits of x to the right by n positions.

summation\_a(a): Returns the result of applying the "summation\_a" operation to the input value a.

summation\_e(e): It provides the result of applying the "summation\_e" operation to the input value e.

sigma\_0(word): Computes the sigma-0 operation on a 64-bit word.

sigma\_1(word): Calculates the sigma-1 operation on a 64-bit word.

addition\_modulo\_2\_64(value): Performs modular addition of two 64-bit values, ensuring the result remains a 64-bit value.

pad\_message(message): This function takes an input message, adds padding to meet SHA-512 requirements, and appends the message's length in bits to the end.

divide\_to\_blocks(message): Splits the padded message into 1024-bit blocks and provides a list of these blocks.

compression\_function(message): This function applies SHA-512 compression to a 1024-bit message block. It updates the HASH\_VALUE list with the new hash values computed from the block.

**Main code:**

In the main section, the code takes a message input from the user, encodes it in UTF-8, and pads it.

It breaks the padded message into 1024-bit blocks.

The code calculates and displays the message length in bits.

It iterates through the blocks, applying the SHA-512 compression function to each block and updating the hash values.

Finally, the code converts and displays the final SHA-512 hash as a hexadecimal string.

This code illustrates the essential components of the SHA-512 algorithm, including message preparation, message scheduling, bitwise operations, and hash value updates. It serves as a simplified representation of the algorithm for educational purposes.