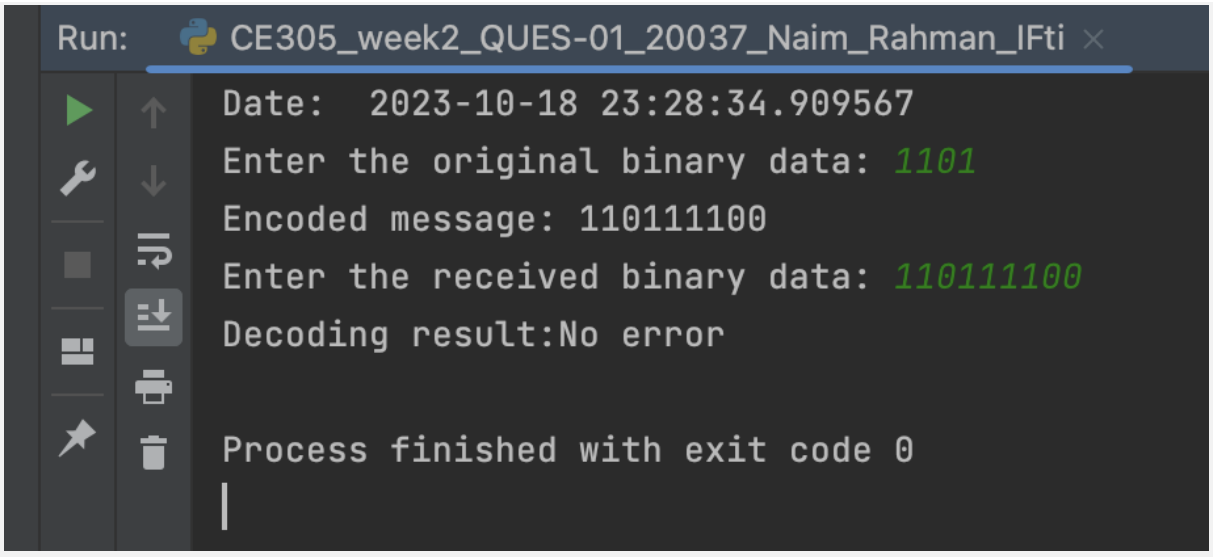
1. Cyclic Redundancy Check (CRC) is one of the popular coding and decoding techniques in the data transmitted over the network for error detection and correction. Given $x^5 + x^2 + 1$ as a CRC generation polynomial from International Telegraph and Telephone Consultative Committee (CCITT), write the encoding and decoding *def* functions in Python for the only 4-bits original binary data. The examples and testcases of the encoding and decoding processes are shown as follows for your programming. After that, discuss how many bits errors CRC can detect.

https://github.dev/dumbnaim/Assignment-02#assignment-02



The number of bit errors a Cyclic Redundancy Check (CRC) can detect is directly

proportional to the size of the CRC in bits. For instance, a CRC-32 can detect up to 32 bits of errors, while a CRC-16 can detect up to 16 bits of errors. Smaller CRCs like CRC-8 can detect up to 8 bits of errors. CRC is a widely used method for error detection in data communication and storage systems. However, it doesn't correct errors but is effective at identifying common errors, making it essential for ensuring data integrity, particularly for single-bit errors and many burst errors.

2. Hamming code is one important error correcting code in computer science and telecommunication as well. Standard Hamming code can only detect and correct a **single bit** error. Encoding method is shown by an example as follows.

https://github.dev/dumbnaim/Assignment-02#assignment-02

-