**Reading the input message from the user keyboard. Encodes in binary Huffman encoding type and then saves the prefix to file. Then retrieves the prefix encoding from the file and uses it to decode the message. Then prints to the screen the decoded message and the prefix encoding.**

**The bolded ones are user inputs.**

Enter the type of demo you want to try (in integer format):

1. Read from User

2. Read from File

3. Save to File

4. Print to Screen

5. Encode Decode

6. Default Easy Encoder

7. Custom Encoder

8. Default Decoder

0. Exit

**5**

Enter the input message to be encoded:

**i am iron man**

Enter the type of huffman encoding (in String Format):

1. Binary

2. Hexadecimal

**binary**

Encoded message: 101011001100110100100011101110100111

Writing the encoded message to disk...at res/runs/encoded\_text.txt

Reading the encoded message from disk...res/runs/encoded\_text.txt

Prefix encoding dictionary: { =01, a=100, r=001, i=101, m=110, n=111, o=000}

Writing the prefix encoding dictionary to disk...at res/runs/prefix\_encoding.txt

Reading the prefix encoding dictionary from disk...res/runs/prefix\_encoding.txt

Prefix encoding dictionary from disk: { =01, a=100, r=001, i=101, m=110, n=111, o=000}

Decoded Message: i am iron man