King Fahd University of Petroleum & Minerals
Department of Electrical Engineering
EE430 Information Theory and Coding
Dr. Ali Muqaibel
Computer Assignment II: Compression (5%)

Due: Week 10/ Class 1
As an introduction to the assignment, make sure that you understand the material presented in the class and handout (Huffman & the LZ code).

You are required to build four Matlab **functions** to implement Huffman & Lemple-Ziv encoders and decoders. The encoder Matlab function should accept text file as the input, convert the file into its binary equivalent and then apply algorithm, generate a new compressed file. The other function should accept the compressed file and generate the original file. It should work for any text file. You may test your program for different files and compare its performance with the commercial WINZIP® or PKZIP® software. Use the provided/own examples to draw different conclusions.

Function [compressed_file]=**Huffman_encode**(uncompressed_file)
Function [uncompressed_file]=**Huffman_decode**(compressed_file)
Function [compressed_file]=**LZ_encode**(uncompressed_file)
Function [uncompressed_file]=**LZ_decode**(compressed_file)

Some of the commands that you might need: *importdata, double, char, dec2bin,….*
Major Steps:

1. Read the file, convert characters into ASCII, convert ASCII code into binary.
2. Apply the LZ code, a) consider what happen if the table is full. b) find a solution for the end of file !, Save the compressed file. In case of Huffman encoding, consider selecting the proper codeword width.
3. Decompress.
4. Convert the binary into ASCII, then convert the ASCII code into characters, save the uncompressed file.
5. Possible extension is to make the program work for picture files, sound,…etc
6. Study the performance of your algorithm for the given files.

You need to submit a total of **no more than 12 pages** including the code. A hard copy should be submitted in the class on the due date and a soft copy (report+m-file) should be submitted through e-mail earlier. If you fail to make it work for text files, start with a long binary data and show that you can compress and decompress the stream.

Instructions:

1. Writing style and organization are very important (Quality not Quantity!)
2. **Your names and serial numbers should be clearly presented on the first page as well as on the code.**
3. **You should work** in a group of 5 students, submit one report, indicate who is responsible for the decoder /encoder. (project leader, H. Encoder + H. Decoder, LZ. Encoder + LZ. Decoder)
4. You should make your output clear and nice. If you have plots, use commands like (axis, xlabel, ylabel, title, legend, text)
5. Remember to use (**help**, **lookfor**) commands.
6. This assignment accounts for **5%** of your total grade.
7. Allow yourself enough time. Do not work close to the due date.
8. Projects are to be submitted during class time. Any late submission will result in zero or low grade.
9. You can use the discussion group in the Course WebCT to discuss general ideas and questions.
10. **Copying is the easiest way to loose points**. To avoid copying algorithms from the internet, you have to implement the algorithm presented in the class. You should write the code yourself and be ready to answer any question related to it.