Name: Vivek More **Batch:** M13

Roll no: 35141

```
%=================================================================
% Experiment No. 10: To Implementation of Huffman codes by using suitable software.
%=================================================================

% --- Initial Setup and Input Data ---
clc;
clear all;
close all;

symbols = {1, 2, 3, 4, 5, 6};
probabilities = [0.3, 0.25, 0.2, 0.12, 0.08, 0.05];
entropy = -sum(probabilities .* log2(probabilities));

%=================================================================
% Huffman Coding Implementation
%=================================================================
numeric_symbols = 1:length(symbols);
huffman_dict = huffmandict(numeric_symbols, probabilities);
huffman_codes = huffman_dict;
huff_lengths = cellfun(@length, huffman_codes);
huff_avg_length = sum(probabilities .* huff_lengths);
huff_efficiency = entropy / huff_avg_length;
huff_redundancy = 1 - huff_efficiency;

%=================================================================
% --- Display Final Results ---
```

```
%==========================================================================
fprintf('=========================================================\n');
fprintf('          Huffman Code Results\n');
fprintf('=========================================================\n');
disp('Generated Codes:');
for i = 1:length(symbols)
    code_as_string = strrep(num2str(huffman_codes{i}), ' ', '');
    fprintf('  Symbol x%d (p=%.2f): %s\n', symbols{i}, probabilities(i), code_as_string);
end
disp('---------------------------------------------------');
fprintf('1. Source Entropy (H)       : %.4f bits/symbol\n', entropy);
fprintf('2. Average Code Length (L)  : %.4f bits/symbol\n', huff_avg_length);
fprintf('3. Code Efficiency (eta)    : %.4f or %.2f%%\n', huff_efficiency, huff_efficiency *
100);
fprintf('4. Redundancy (gamma)       : %.4f or %.2f%%\n', huff_redundancy, huff_redundancy
* 100);
fprintf('=========================================================\n');
```

OUTPUT: