```
% Created on 14/02/25
% Created by Saurav kumar, BT22ECE110
function [symbols_codes, symbols_codewords] = Experiment_4()
    clc;
    clear;
    close all;

    % Define the symbols and their respective probabilities
    symbols = ['A', 'B', 'C', 'D', 'E'];
    probabilities = [0.2, 0.05, 0.5, 0.2, 0.05];

    % Ensure probabilities sum to 1 (normalize if necessary)
    if sum(probabilities) ~= 1
        probabilities = probabilities / sum(probabilities);
    end

    num_symbols = length(symbols);
    symbol_data = cell(num_symbols, 2);

    % Store symbols and their probabilities in a cell array
    for i = 1:num_symbols
        symbol_data{i, 1} = symbols(i);
        symbol_data{i, 2} = probabilities(i);
    end

    % Sort symbols based on probability in descending order
    symbol_data = sortrows(symbol_data, -2);

    % Generate Shannon-Fano codes
    [symbols_codes, symbols_codewords] =
generate_shannon_fano_code(symbol_data, '');

    % Display the generated codes
    fprintf('Shannon-Fano Codes:\n');
    for i = 1:length(symbols_codes)
        fprintf('Symbol: %s, Code: %s\n', symbols_codes{i},
symbols_codewords{i});
    end
end

function [codes, codewords] = generate_shannon_fano_code(symbol_data, prefix)

    num_symbols = size(symbol_data, 1);

    % Base case: If only one symbol remains, assign the prefix as its code
    if num_symbols == 1
        codes = {symbol_data{1, 1}};
        codewords = {prefix};
        return;
    end

    % Compute cumulative probabilities to find the best split point
    cumulative_probabilities = cumsum([symbol_data{:, 2}]);
    [~, split_index] = min(abs(cumulative_probabilities -
cumulative_probabilities(end) / 2));

    % Split symbols into two partitions based on probability distribution
    left_partition = symbol_data(1:split_index, :);
    right_partition = symbol_data(split_index+1:end, :);

    % Recursively assign binary codes to each partition
    [left_codes, left_codewords] = generate_shannon_fano_code(left_partition,
[prefix '0']);
    [right_codes, right_codewords] =
generate_shannon_fano_code(right_partition, [prefix '1']);

    % Combine results from left and right partitions
    codes = [left_codes, right_codes];
    codewords = [left_codewords, right_codewords];
end
```

```
Shannon-Fano Codes:
Symbol: C, Code: 0
Symbol: A, Code: 10
Symbol: D, Code: 110
Symbol: B, Code: 1110
Symbol: E, Code: 1111

ans =

  1x5 cell array

    {'C'}    {'A'}    {'D'}    {'B'}    {'E'}
```

Published with MATLAB® R2024b