Table of Contents

Arithmetic Coding – Alternate Example 1
Symbol Set and Probabilities 1
Uncomment for interactive input 1
Compute cumulative distribution 1
Encoding Loop 1
Encoded Result 2

Arithmetic Coding – Alternate Example

Same algorithm, different data & coding style

```
clc;
clear;
close all;
```

Symbol Set and Probabilities

```
symbols = ["X", "Y", "Z", "W"];
probabilities = [0.1 0.4 0.3 0.2];
message = 'XYZWY';
```

Uncomment for interactive input

symbols = input("Enter symbol array: "); probabilities = input("Enter probability array: "); message = input("Enter message string: ", 's');

Compute cumulative distribution

```
cumulativeProb = [0, cumsum(probabilities)];

lowLimit = 0;
highLimit = 1;

fprintf('Initial interval: [%.6f , %.6f)\n\n', lowLimit, highLimit);
```

```
Initial interval: [0.000000 , 1.000000)
```

Encoding Loop

```
for idx = 1:length(message)

    % Identify symbol position
    pos = find(symbols == message(idx));

    % Current interval width
    intervalSize = highLimit - lowLimit;

    % Update interval bounds
    lowTemp = lowLimit + intervalSize * cumulativeProb(pos);
    highTemp = lowLimit + intervalSize * ...
               (cumulativeProb(pos) + probabilities(pos));

    % Assign updated bounds
    lowLimit = lowTemp;
    highLimit = highTemp;

    fprintf('After %c → [%.6f , %.6f)\n', message(idx), lowLimit, highLimit);
end
```

```
After X → [0.000000 , 0.100000)
After Y → [0.010000 , 0.050000)
After Z → [0.030000 , 0.042000)
After W → [0.039600 , 0.042000)
After Y → [0.039840 , 0.040800)
```

Encoded Result

```
fprintf('\nEncoded interval for "%s": [%.6f , %.6f)\n', ...
        message, lowLimit, highLimit);

encodedNumber = (lowLimit + highLimit) / 2;
fprintf('Final encoded value: %.6f\n', encodedNumber);
```

```

Encoded interval for "XYZWY": [0.039840 , 0.040800)
Final encoded value: 0.040320
```

Published with MATLAB® R2025b