Problem 1

As in HW 5, again consider the restricted character space of the 14 uppercase English letters:

{A, B, C, D, E, F, G, H, I, M, N, R, S, T}

with associated probability distribution given by:

A	B	C	D	E	F	G	H	I	M	N	R	S	T
0.1061	0.0194	0.0362	0.0556	0.1643	0.0285	0.0259	0.0789	0.0906	0.031	0.0867	0.0776	0.0815	0.1177

a) In HW 5 you computed the entropy of the probability distribution for this character set. Based on that result, what would you predict as the lower bound on average bit length for an efficient, variable bit length binary encoding of this character set?

Entropy measures the average amount of information from identifying the outcome of a random trial. The log base 2 gives units of bits.

$$H(\{p_i\}) = -\sum_i p_i \log_2 p_i = -(0.1061 \log_2 0.1061 + \cdots + 0.1177 \log_2 0.1177)$$

$$= 3.5782 \text{ bits}$$

* HW5 solution

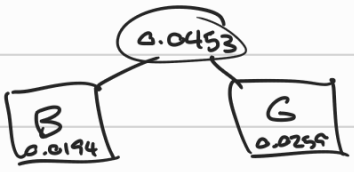
b) Apply the graph/tree construction algorithm to create a Huffman encoding for this character set, with associated probabilities. Employ the convention illustrated in lecture, to select the most significant bit of the binary representation based on the "highest" branches of the tree, and 0 bits assigned for "left branches," 1 bits assigned for right branches.

There is a very nice animation in this link that illustrates how the tree is built:

https://opendsa-server.cs.vt.edu/ODSA/Books/CS3/html/Huffman.html

Start by ordering the probabilities in ascending order.

Then, combine the two nodes with the lowest probability.



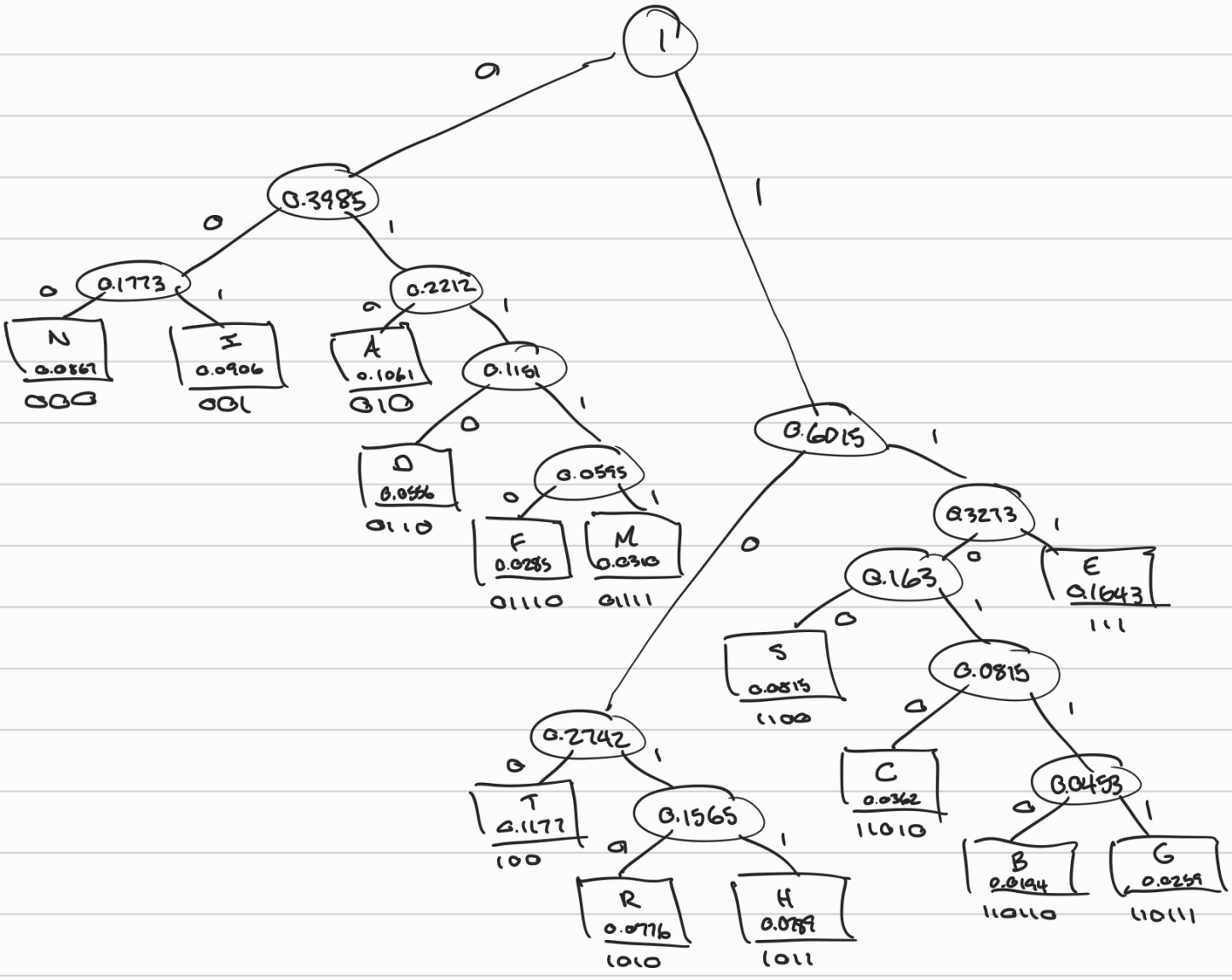
After combining, re-sort the combined node with the rest of the nodes.

```
freq_sorted =

  14×2 cell array

    {'B'}    {[0.0194]}
    {'G'}    {[0.0259]}
    {'F'}    {[0.0285]}
    {'M'}    {[0.0310]}
    {'C'}    {[0.0362]}
    {'D'}    {[0.0556]}
    {'R'}    {[0.0776]}
    {'H'}    {[0.0789]}
    {'S'}    {[0.0815]}
    {'N'}    {[0.0867]}
    {'I'}    {[0.0906]}
    {'A'}    {[0.1061]}
    {'T'}    {[0.1177]}
    {'E'}    {[0.1643]}
```

The resulting tree should look like the one below:



c) What is the expected bit length for this encoding, assuming messages satisfying the character probability distribution as given?

Average length of codeword given by: $l_{avg} = \sum_i l_i \cdot p_i$

$$l_{avg} = 6(0.0194 + 0.0259) + 5(0.0285 + 0.0310 + 0.0362)$$
$$+4(0.0556 + 0.0776 + 0.0789 + 0.0815) + 3(0.0867 + 0.0906 + 0.1061 + 0.1177 + 0.1643)$$

$$= 3.6209 \text{ bits}$$

d) Does this construction guarantee that the binary representation for each character produced "unique prefixes?" For example, consider the characters that have the shortest bit length encoding; denote this shortest bit length as m. Do any characters that have longer bit length encodings begin with same first m-bits as any of the m-length encoded characters? Justify your conclusion based on the structure of the tree. Would this conclusion change if the most significant bit were assigned based on the "lowest" branches of the tree, and least significant bit based on the highest branches?

Looking at the 3-bit encodings of the symbols on the tree, none of these codes appear as prefixes for each of the longer bit encodings. This construction (Huffman coding) guarantees unique prefixes for the binary representation of each character.

If the most significant bit were assigned based on the "lowest" branches and the least significant bits, this conclusion would change. (Consider the case with E and M).

Problem 2

Suppose that we are encrypting plaintext English language messages using the 96 printable ascii characters (that may be encoded as $\mathbf{Z}_{96}$). The encryption is performed using a periodic Vigenere cipher of length m. You may use our standard assumption for entropy value of English language messages; that is, H=2.0.

a) What is the numeric value for redundancy R for this scenario?

Redundancy (from Bach's notes) given by: $R = \log_2 N - H$

$$R = \log_2 96 - 2.0 = 4.5849$$

b) Based on the lower bound estimate examined in the Bach notes and lecture, express the unicity point for this scenario as a function of period length m.

Estimate on 15-3 of Bach's notes

$$u = \frac{m \cdot \log_2 96}{R} = \frac{m \cdot \log_2 96}{\log_2 96 - 2.0} = 1.4362m$$

Problem 3

Consider linear congruential generators (lcg's) for pseudo-random key streams in the case of keys on $\mathbf{Z}_{96}$.

a) Consider a lcg having coefficients a=7 and b =22. For key streams on $\mathbf{Z}_{96}$ produced by this lcg, over any possible seed key k_0, what is the shortest period and what is the longest period? (suggestion: identify a first periodic stream starting from key k_0 =0; your second string can then begin from smallest seed that was NOT among the values appearing in the first string; your third string can then begin from smallest seed that was NOT among the values appearing in the first and second string; etc.)

Can be done by testing many values for k_0, then observing the period.
I did this using MATLAB (see hw6p3sol.m and lcg.m)

```
max_period =
    12

k0_max =
     2

min_period =
     3

k0_min =
     7


>>
```

hw6p3sol – tests many k_0 for given lcg parameters

lcg – recursive algorithm that implements a linear congruential generator with basic error checking

b) Identify coefficients "a" and "b" for a lcg that produces a periodic stream of period 96 (i.e., the maximum possible period N). To standardize your answer, identify the smallest possible "a" and smallest possible "b" that produce this result.

This is the Canvas result.

One such choice is a=b=1 (Bach's notes, 18-2). While this does produce a key stream of period 96, we'll see a more interesting solution shortly.

```
>> [k,m]=lcg(1,1,96,0)

k =

  Columns 1 through 23

     0     1     2     3     4     5     6     7     8     9    10    11    12    13    14    15    16    17    18    19    20    21    22

  Columns 24 through 46

    23    24    25    26    27    28    29    30    31    32    33    34    35    36    37    38    39    40    41    42    43    44    45

  Columns 47 through 69

    46    47    48    49    50    51    52    53    54    55    56    57    58    59    60    61    62    63    64    65    66    67    68

  Columns 70 through 92

    69    70    71    72    73    74    75    76    77    78    79    80    81    82    83    84    85    86    87    88    89    90    91

  Columns 93 through 97

    92    93    94    95     0

m =

    96

>>
```

0. Find 2 distinct prime factors of N = 96.

$96 = 2^5 \cdot 3 \longrightarrow p_1 = 2;\ p_2 = 3$

1. Choose b to be relatively prime to N.

Notice that b need not be prime itself. For b to be relatively prime to N, b must not share common prime factors with N.

b=1 satisfies this requirement.

2. Choose a such that:

i) every p_i divides (a-1)

ii) if 4 is an even divisor of N, then it must also divide (a-1) evenly

The smallest (a-1) that satisfies i) and ii) is:

$(a-1) = 4 \cdot 3 = 12$ ... so the smallest a = 13

Below is a sequence produced by this lcg with $k_0 = 1$:

```
>> [k,m]=lcg(13,1,96,1)

k =

  Columns 1 through 23

     1    14    87    76    29    90    19    56    57    70    47    36    85    50    75    16    17    30     7    92    45    10    35

  Columns 24 through 46

    72    73    86    63    52     5    66    91    32    33    46    23    12    61    26    51    88    89     6    79    68    21    82

  Columns 47 through 69

    11    48    49    62    39    28    77    42    67     8     9    22    95    84    37     2    27    64    65    78    55    44    93

  Columns 70 through 92

    58    83    24    25    38    15     4    53    18    43    80    81    94    71    60    13    74     3    40    41    54    31    20

  Columns 93 through 97

    69    34    59     0     1

m =

    96
```