

Activity: Huffman Coding

Part 1: Use the *Huffman codes* to decode the following binary sequence (go from left-to-right):

Code	00	0100	0101	0110	0111	100	101	110	1110	1111
Char	E	P	N	G	H	O	B	T	I	S

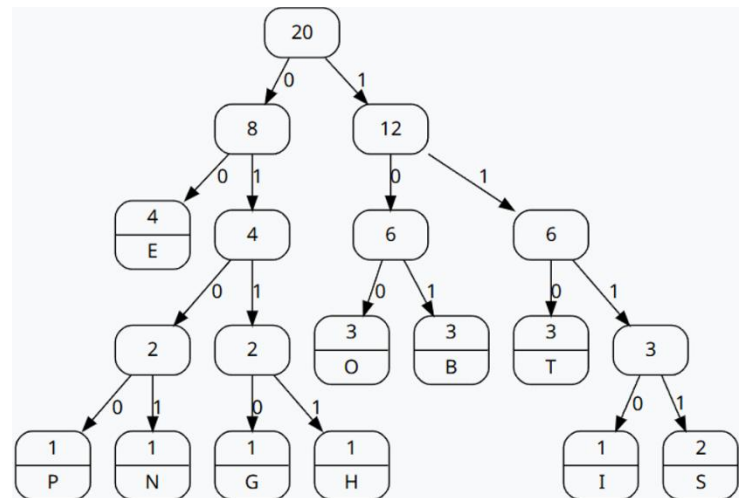
Binary: 010010001010110100101001100111001010011111100011111101011001110

Decoded: _____

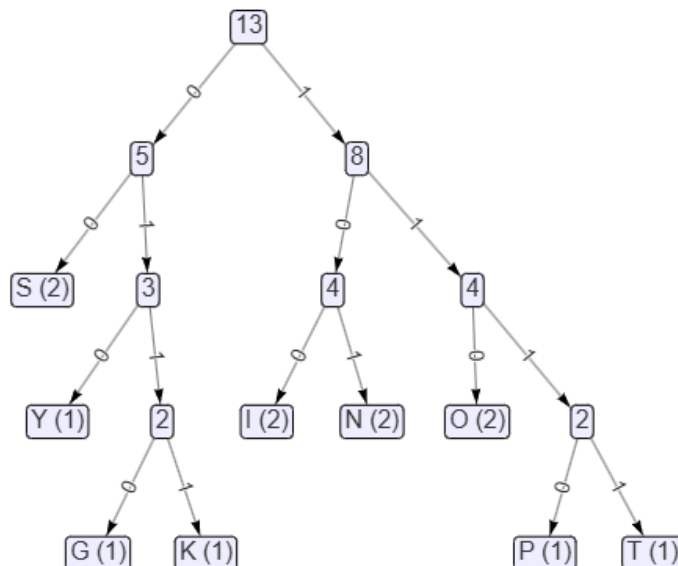
Consider the *Huffman tree* on the right:

- Each **leaf** contains a letter and a frequency.
- Each **internal node** contains the sum of the frequencies of its children.
- This **Huffman tree** was used to generate the codes above.

How were the binary codes for each character determined using the tree?



Part 2: Use the following Huffman tree to decode the binary sequence below:



Encoded: 11101101010110110100000011111001010111010

Decoded: _____

Part 3: Consider this string of characters:

EEPYPUPPY

1. Determine **frequency**
2. Make **priority queue** with nodes
3. Make **subtree** from two smallest
4. Put subtree **back** in queue
5. Repeat **3 & 4** until one item left

1. Create a frequency table:

<i>Character</i>	<i>Frequency</i>

2. Now, create a Huffman tree based on the frequency table.

(Break ties alphabetically – if two subtrees have the same frequency, the one containing the letter that is earliest in the alphabet should come first.)

Work through step-by-step:

3. Create a mapping of character to binary code:

<i>Character</i>	<i>Code</i>

4. Encode the original string to binary:

EEPYPUPPY

5. How many bits were required to encode the string? _____