CMSC 204

Huffman Lab

1. Create a Huffman Tree and generate the codes for each character of the following input:

create a huffman tree

For consistency:

1. If same frequency – put in priority queue alphabetically; put space before other characters of the same frequency
2. Add subtrees to end of group with same priority
3. Lower number has higher priority (goes to front)

|  |  |
| --- | --- |
| Character | Frequency |
| space | 3 |
| c | 1 |
| r | 2 |
| e | 4 |
| a | 3 |
| t | 2 |
| h | 1 |
| u | 1 |
| f | 2 |
| m | 1 |
| n | 1 |

Sort with respect to frequency

|  |  |
| --- | --- |
| c, h, u, m, n | 1 |
| r, t, f | 2 |
| space, a | 3 |
| e | 4 |

Now encode “create a huffman tree”

c r e a t e [Sp] a [Sp] h u f f

0100 1110 110 100 000 110 101 100 101 0101 0110 001 001

m a n [sp] t r e e

0111 100 1111 101 000 1110 110 110

Code:

0100 1110 110 100 000 110 101 100 101 0101 0110 001 001 0111 100 1111 101 000 1110 110 110

Which is:

0100111011010000011010110010101010110001001011110011111010001110110

1. Based on the following Huffman tree and binary sequence, what is the text



**0**

**0**

**0**

**0**

**0**

**0**

**0**

**0**

**0**

**1**

**1**

**1**

**1**

**1**

**1**

**1**

**1**

**1**

1110011101101111111010001100010001100100

|  |  |
| --- | --- |
| **Character codes:** | |
| n | 000 |
| r | 001 |
| t | 010 |
| u | 011 |
| e | 100 |
| f | 101 |
| space | 1100 |
| a | 1101 |
| h | 1110 |
| m | 1111 |

1110011101101111111010001100010001100100 =

1110 011 101 101 1111 1101 000 1100 010 001 100 100

h u f f m a n space t r e e

Message: “huffman tree”