Name: Vaishnavi Someshwar Gawande

Roll No: BEA 29

Assignment No: 02

Problem Statement: Write a program to implement Huffman Encoding using a greedy strategy

Code:

```
import heapq

class node:
    def __init__(self, freq, symbol, left=None, right=None):
        self.freq = freq
        self.symbol = symbol
        self.left = left
        self.right = right
        self.huff = ''

    def __lt__(self, nxt):
        return self.freq < nxt.freq

    def printNodes(node, val=''):
        newVal = val + str(node.huff)
        if(node.left):
            printNodes(node.left, newVal)
        if(node.right):
            printNodes(node.right, newVal)
        if(not node.left and not node.right):
            print(f"{node.symbol} =>  {newVal}")
```

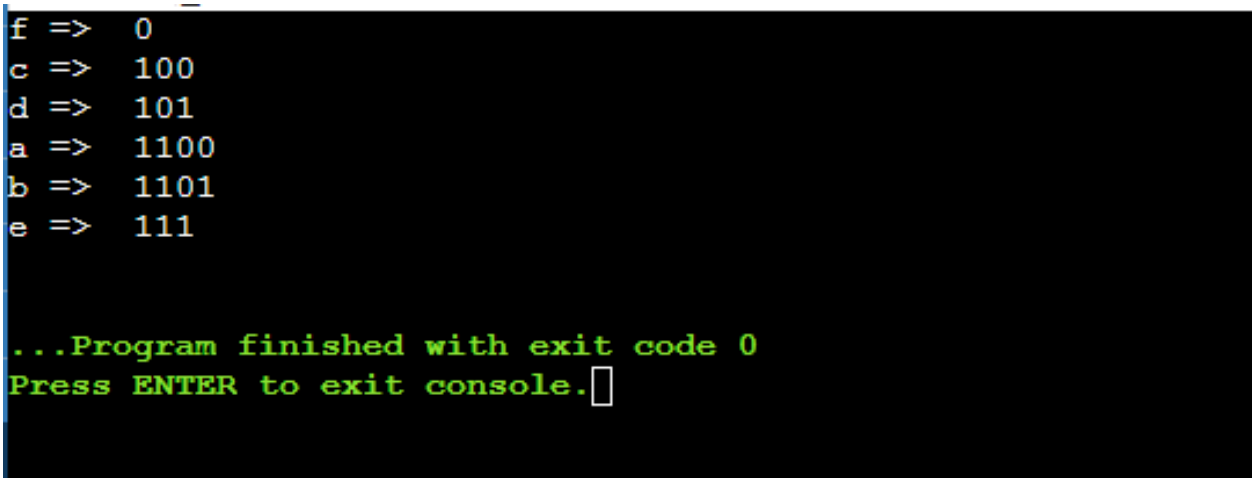
```
chars = ['a', 'b', 'c', 'd', 'e', 'f']
freq = [ 5, 9, 12, 13, 16, 45]
nodes = []
for x in range(len(chars)):
    heapq.heappush(nodes, node(freq[x], chars[x]))

while len(nodes) > 1:
    left = heapq.heappop(nodes)
    right = heapq.heappop(nodes)
    left.huff = 0
    right.huff = 1
    newNode = node(left.freq+right.freq, left.symbol+right.symbol, left, right)
    heapq.heappush(nodes, newNode)
    printNodes(nodes[0])
```

Input:

```
chars = ['a', 'b', 'c', 'd', 'e', 'f']
freq = [ 5, 9, 12, 13, 16, 45]
```

Output:


f => 0
c => 100
d => 101
a => 1100
b => 1101
e => 111
...Program finished with exit code 0
Press ENTER to exit console.