Fixed vs Variable Length Encoding

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency (in thous)</td>
<td>45</td>
<td>13</td>
<td>12</td>
<td>16</td>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td>Fixed-length codeword</td>
<td>000</td>
<td>001</td>
<td>010</td>
<td>011</td>
<td>100</td>
<td>101</td>
</tr>
<tr>
<td>Variable length codeword</td>
<td>1</td>
<td>011</td>
<td>010</td>
<td>000</td>
<td>0011</td>
<td>0010</td>
</tr>
</tbody>
</table>

Character Encoding:

- 100,000 character file which contains only the characters A-F
- Frequencies of each character appear (in thousands)

Questions:

1) Without any encoding how many bits make up this file?

2) Assuming a fixed length coding:
   a) Why in this example is three bits used as the fixed length code?
   b) How many bits now make up the file?

3) Assuming a variable length coding:
   a) Why use a variable length coding?
   b) How many bits make up the file now?
Huffman Tree

```
        100
       /  
      55   1
     /  \
    30   25
   /  \
  16   12
 /   /  \
D   14   C
 /   /  \
F   E   B
```

**CLASS EXERCISE**

**DECODE 54 OC USING LEFT-TO-RIGHT DECISION**

**WORD IS FACED**

```
1) HUFFMAN TREE IS NOT UNIQUE (CLASS EXAMPLE)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>F(g)</th>
<th>Bit Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>B</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>C</td>
<td>9</td>
<td>1</td>
</tr>
<tr>
<td>D</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>EOF</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
```

**Encode (ADCBACCEOF)**