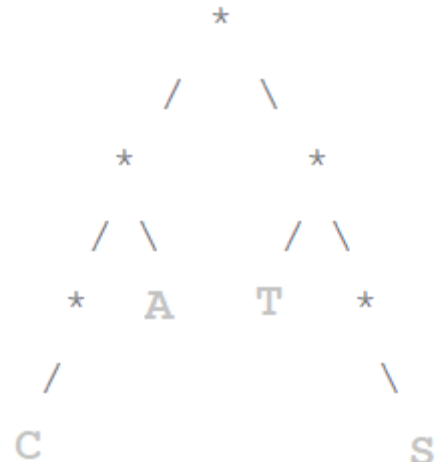


# Huffman code

This problem was asked by Amazon

Huffman coding is a method of encoding characters based on their frequency. Each letter is assigned a variable-length binary string, such as `0101` or `111110`, where shorter lengths correspond to more common letters. To accomplish this, a binary tree is built such that the path from the root to any leaf uniquely maps to a character. When traversing the path, descending to a left child corresponds to a `0` in the prefix, while descending right corresponds to `1`.

Here is an example tree (note that only the leaf nodes have letters):



With this encoding, `CATS`, would be represented as `0000110111`.

Amazon asked:

Given a dictionary of character frequencies, build a Huffman tree, and use it to determine a mapping between characters and their encoded binary strings.

Your task here is simpler. In the follow up program we will solve this problem. The follow up problem is the assignment from Harvey Mudd College intro programming class for students who have no programming experience.

Your task is, given an array representation of a binary tree, and a String of 0's and 1's, return the message.

For example, if

```
String[] letterTree = new String[16];
```

```
letterTree[5] = "A";
```

```
letterTree[6] = "T";
```

```
letterTree[8] = "C";
```

```
letterTree[15] = "S";
```

and

```
HuffmanCodeWithArray hc = new HuffmanCodeWithArray(letterTree;
```

then

```
hc.getHuffmanMessage("0000110111");
```

returns the string `"CAT"`.