

1) check1

2) ./compress check1.txt outcome.txt

99 10

101 10

```
111001001110010011100100111001001110010011100100111001001110010011100100
11100100
```

Find

```

graph TD
    Root((40)) -- 1 --> L20((20))
    Root -- 0 --> R20((20))
    L20 -- 1 --> L10a((10))
    L20 -- 0 --> L10b((10))
    R20 -- 1 --> R10c((10))
    R20 -- 0 --> R10d((10))
    L10a --- a[a]
    L10b --- b[b]
    R10c --- c[c]
    R10d --- d[d]
    R10c --- 10c[10]
    R10d --- 10d[10]
  
```

a=11

c=01

d=00

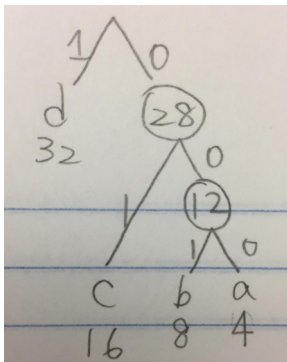
Following the encoded string, we get “11 | 10 | 01| 00 “ repeated 10 times, which is abcd repeated 10 times, the same as the inputted file.

101 32

[illegible]

connect them together to `p(d,cba)(60)` and connect the parent node and children node.

Suppose we want to find the code of a, I find the code by starting from the root **p(d, cba)** and then go through the line leading to a with labels **1 or 0**. The concatenated string with 0 and 1 is the code of the letter in this tree.



d 1

### Efficient Header

We change the size of each frequency from 4 byte to 3 byte which saves 1024 bits. Since we don't have more than  $2^{24}$  number of characters in a file, we don't need so much space to save it.