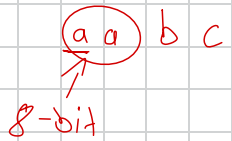
Huffman coding



t t a a b c c t a a c m c a a a m m t t t c

① histogram

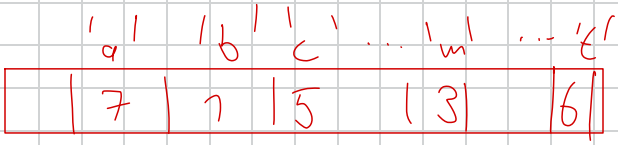
t → 6

a → 7

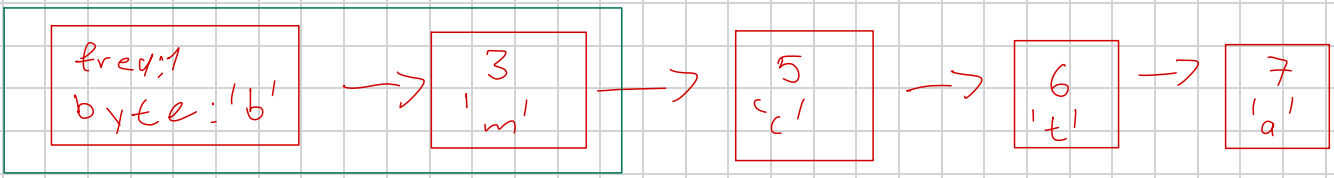
b → 1

c → 5

m → 3

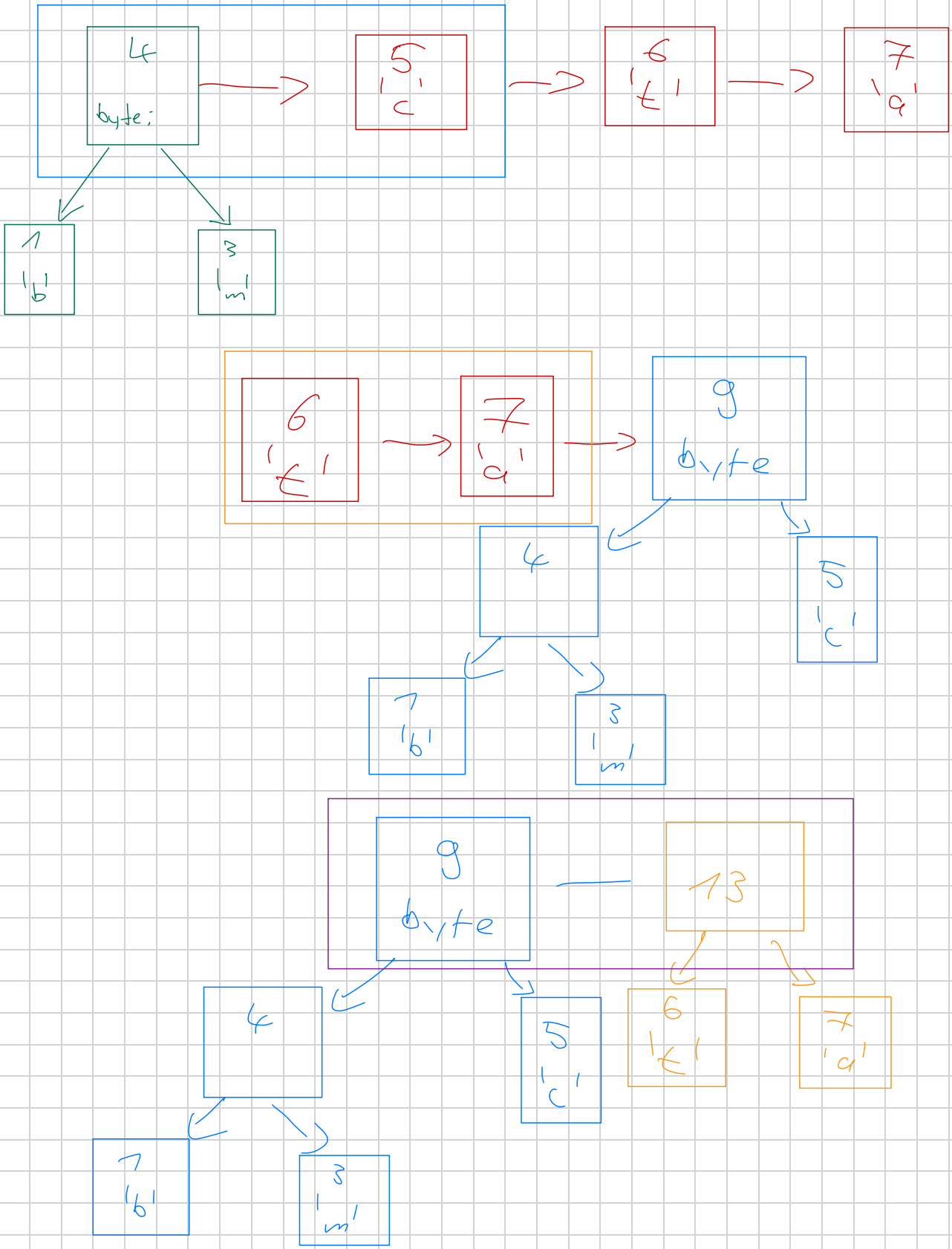


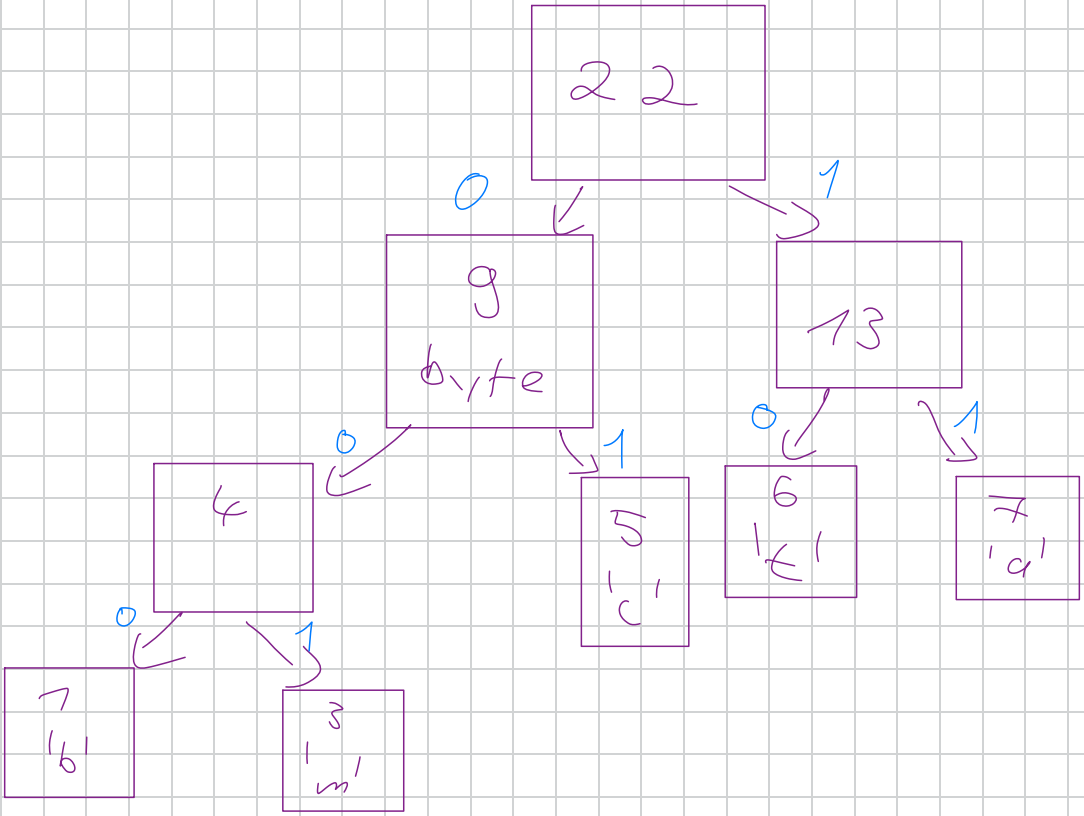
② Build a list




```
struct node {
    int freq;
    char byte;
    struct node *next;
};
```

③ Build a tree as long as there are nodes





'b' → 000

'm' → 001

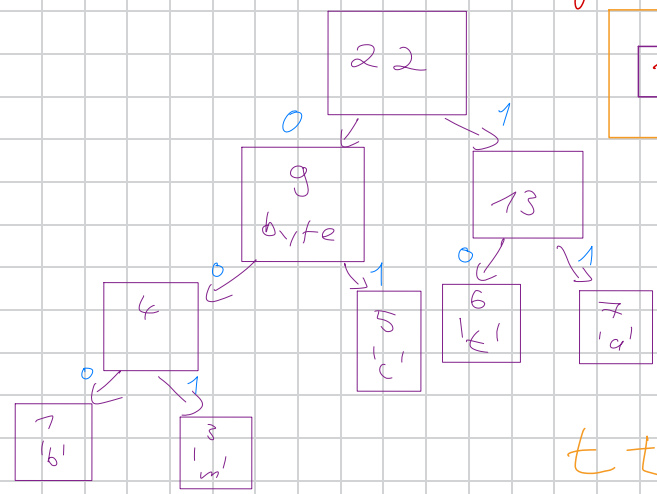
'c' → 01

't' → 10

'a' → 11

④ build encoding t t a a b c c t a a c m c a a a m m t t t c

10 10 11 11 000 01



t t a a b c

① histogram

② list in order of frequency

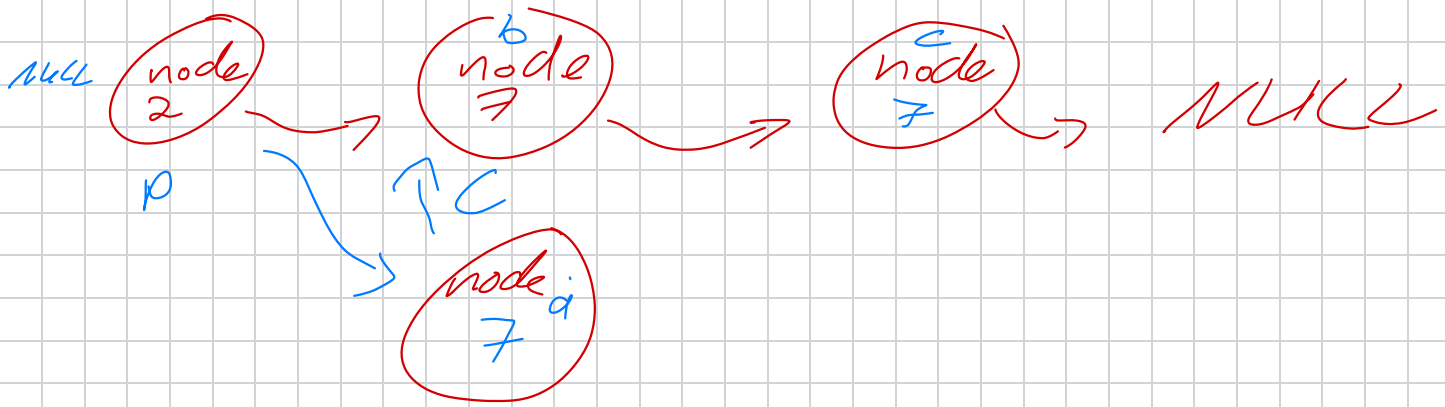
③ tree (list => tree)

④ encoding

⑤ use encoding to compress

⑥

```
struct node {
    int freq;
    char byte;
    struct node *next;
    struct node *left;
    Struct node *right;
};
```


NULL
node
2
b
node
3
c
node
7
NULL
p
c
node
7
a