**Quiz 5**

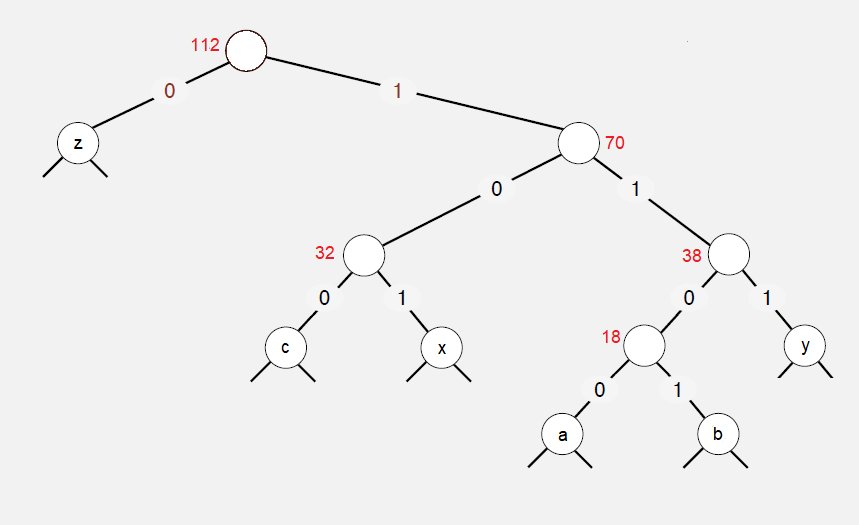
**İbrahim Burak Tanrıkulu 21827852**

**Q1**

Creating tree:

* a and b have minimum frequency. Merge into a tree with 7+11 = 18 frequency.
* c and x have minimum frequency. Merge into a tree with 14+18 = 32 frequency.
* a-b tree and y have minimum frequency. Merge into a tree with 18+20 = 38 frequency.
* c-x tree and (a-b)-y tree have minimum frequency. Merge into a tree with 32+38 = 70 frequency.
* z and (c-x)-((a-b)-y) tree have minimum frequency. Merge into a tree with 42+70 = 112 frequency.

We created our tree as below:



From the tree, we encoded our variables to these values:

z = 0

c = 100

x = 101

y = 111

a = 1100

b = 1101

Number of bits: 112\*16 = 1792

Number of bits with new transmission:

4\*7 + 4\*11 + 3\*14 + 3\*18 + 3\*20 + 1\*42 = 28+44+42+54+60+42 = 270

We saved 1522 bit.

**Q2**

Our string: “rmmnnnntttt”

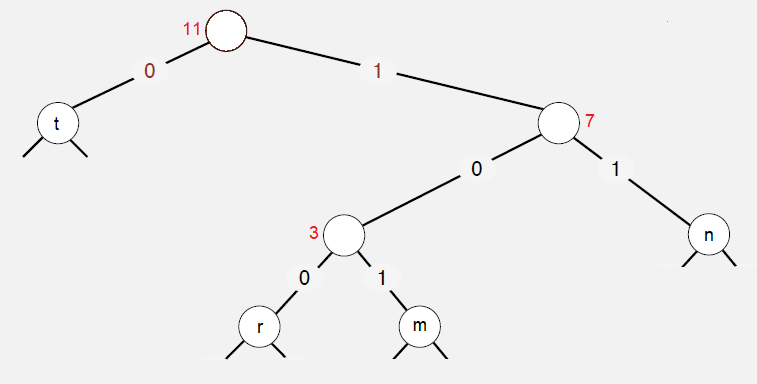
Frequencies:

r = 1 m = 2 n = 4 t = 4

Creating tree:

* r and m have minimum frequency. Merge into a tree with 1+2 = 3 frequency.
* r-m tree and n have minimum frequency. Merge into a tree with 3+4 = 7 frequency.
* (r-m)-n tree and t have minimum frequency. Merge into a tree with 7+4 = 11 frequency.

We created our tree as below:



From the tree, we encoded our variables to these values:

t = 0

n = 11

r = 100

m = 101

We need 1\*3 + 2\*3 + 4\*2 + 4\*1 = 3+6+8+4 = **21** bits to transmit this massage via Huffman encoding.

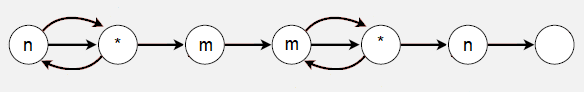
100 101 101 11 11 11 11 0 0 0 0

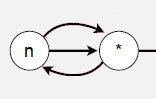
r m m n n n n t t t t

Average bits per character is (1+2+3+3)/4 = 9/4 = **2,25** .

**Q3**

Regular expression: “n\*m+n” or “n\*mm\*n”

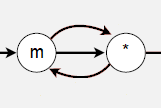




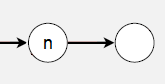
* Firstly, traverse “n” letters if exists.



* Secondly, get first “m” letter.



* Thirdly, traverse “m” letters if exist.



* Finally, get last “n” letter. Thus we found our “mn” substing.