YUQIAN ZHANG #N19945556 Recitation: Ojas

Fundamental Algorithms Problem Set 9

Q1 Huffman Code for $\{v, w, x, y, z\}$ z has 0 or 1.

Prove that the frequency for z cannot be less than $\frac{1}{3}$.

Proof: According to the fact that z has 0 or 1 as the code word. We know that z is greater than any sum of two from $\{v, w, x, y\}$. There are 3 combinations.

$(z \geq v+w)$ and $(z \geq x+y)$

or $(z \geq v+x)$ and $(z \geq w+y)$

or $(z \geq v+y)$ and $(z \geq w+x)$

From above all 3 combinations, we can get the conclusion.

$$\begin{cases} 2z \geq v+w+x+y & ① \\ 1 = v+w+x+y+z & ② \text{ (know as defined)} \end{cases}$$

for ②: $1-z = v+w+x+y$ ③

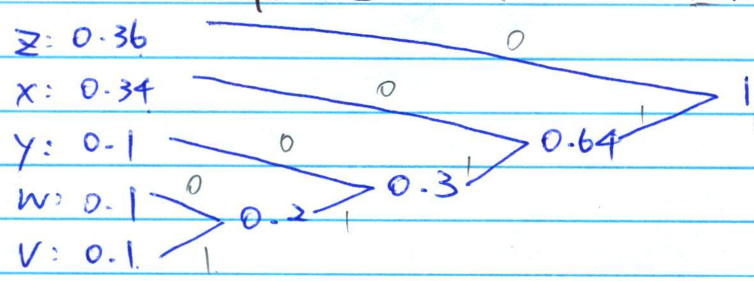
replace ① by ③: $2z \geq 1-z$

$3z \geq 1$

$z \geq \frac{1}{3}$

Thus, we have the conclusion that the frequency for z cannot be less than $\frac{1}{3}$

Give an example z: 0.36 z does ~~not~~ get word 0 or 1

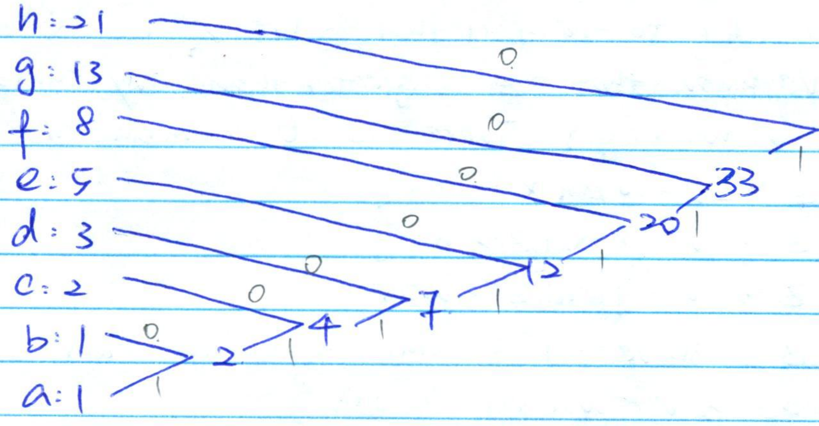


z gets code 0. here

Q2 a:1 b:1 c:2 d:3 e:5 f:8 g:13 h:21

Solution:

(a) Huffman code for the code above:



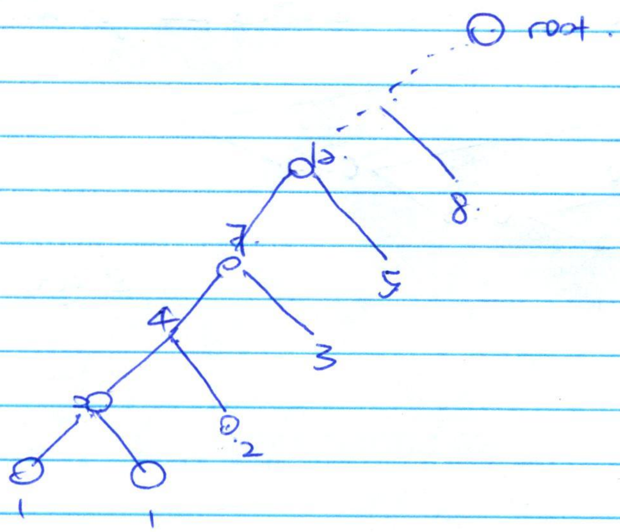
h: 0
g: 10
f: 110
e: 1110
d: 11110
c: 111110
b: 1111110
a: 1111111

Generalie to find the optimal code for n letters Fib number.

(b) The huffman code are:

$0, 10, 110, 1110, 11110, \cdots 1^{n-2}0, 1^{n-1}$

The code tree is a linear structure.
Each level has exactly one leaf, exept the root which has no leaves and the bottom level which has two leaves
The letters with lower frequencies stay at lower level.



Q3. Huffman code Not use Min-heaps. at each step we found the letters of minimal frequency and replaced them by a new letter with frequency their sum.

Solution: $O(n^2)$

assume we have N letters. at each step. (inner loop)

```
for i : 1 to n.                          } O(n)
  a1 = find minimal frequency
for i : 1 to n.                          } O(n)
  a2 = find minimal frequency
create new letter;                         O(1)
new letter.frequency = sum(a1, a2);        O(1)
```

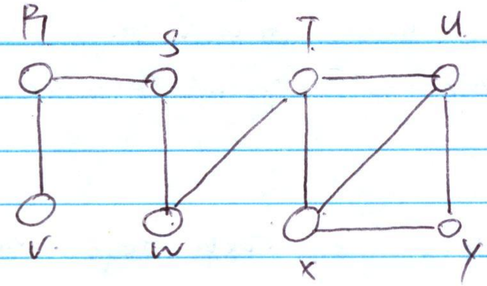
Thus, the complete pseudo code is:

```
for n : N to 1 {
    for i : 1 to n {
        a1 = find minimal frequency }
    for i : 1 to n {                       O(n)
        a2 = find minimal frequency }
    create new letter;
    new letter.frequency = sum(a1, a2);
}
```

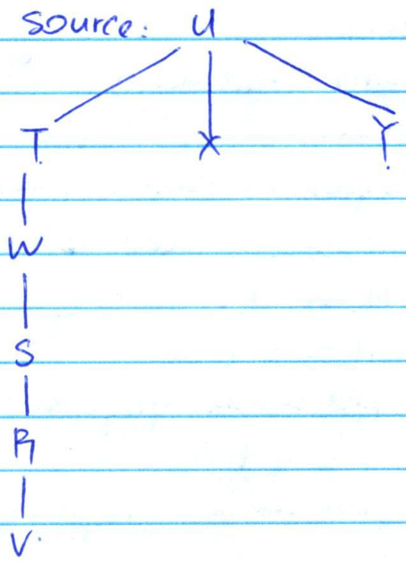
Total time $= N + (N-1) + (N-2) + \cdots$

$= \frac{(N+1)N}{2} \sim O(N^2)$

Q5.



Using vertex u as the source BFS



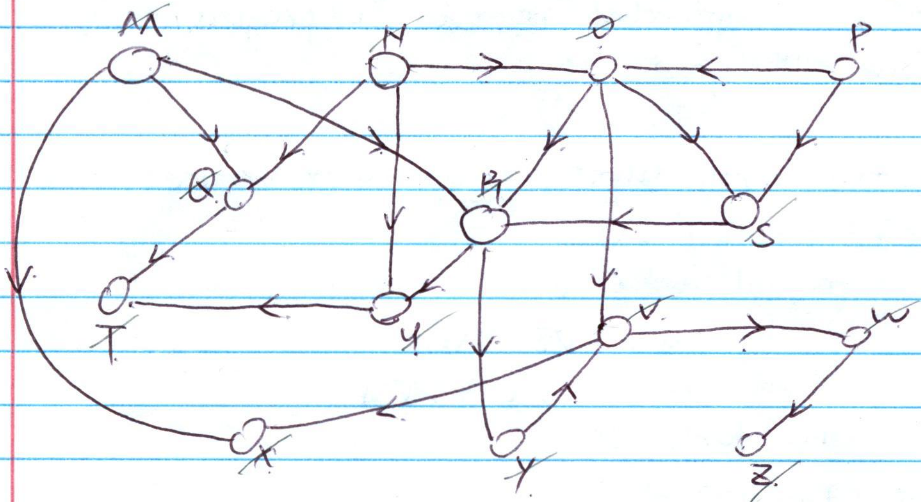
T	$d = 1$	$\pi = u$
X	$d = 1$	$\pi = u$
Y	$d = 1$	$\pi = u$
W	$d = 2$	$\pi = T$
S	$d = 3$	$\pi = W$
R	$d = 4$	$\pi = S$
V	$d = 5$	$\pi = R$

Q 6. Colors : white black grey
untouched processed in progress
Types : good bad

Initialization : white assigned to all vertices

```
BFS - MASTER [G]
    for all v ∈ V
        if (color[V] = white)
            do BFS [G, V];
    end for
BFS [G, V]
color [V] = grey;
type [V] = good;
Queue Q initially Q = (s)
while Q ≠ φ
        u ← dequeue [Q]
        for v ∈ Adj [u]
            if color [V] = white
                color [V] = grey
                if type [u] = good
                    type [V] = bad
                else if type[u] = bad
                    type [V] = good
                enqueue [V]
                π [V] = u
                d [V] = d[u]+1
        end for
        Color [u] ← black
```

Q5. Show the ordering of the vertices produced by TOPSORT



Solution:

		start time	finish time	Finally returns
5	M	1	~~20~~	P
2	N	21	~~26~~	N
3	O	22	~~25~~	O
1	P	27	~~28~~	S
13	Q	2	~~5~~	M
6	R	6	~~19~~	R
4	S	23	~~24~~	Y
14	T	3	~~4~~	V
12	U	7	~~8~~	X
8	V	10	~~17~~	W
10	W	11	~~14~~	Z
9	X	15	~~16~~	U
~~7~~	Y	9	~~18~~	Q
11	Z	12	~~13~~	T