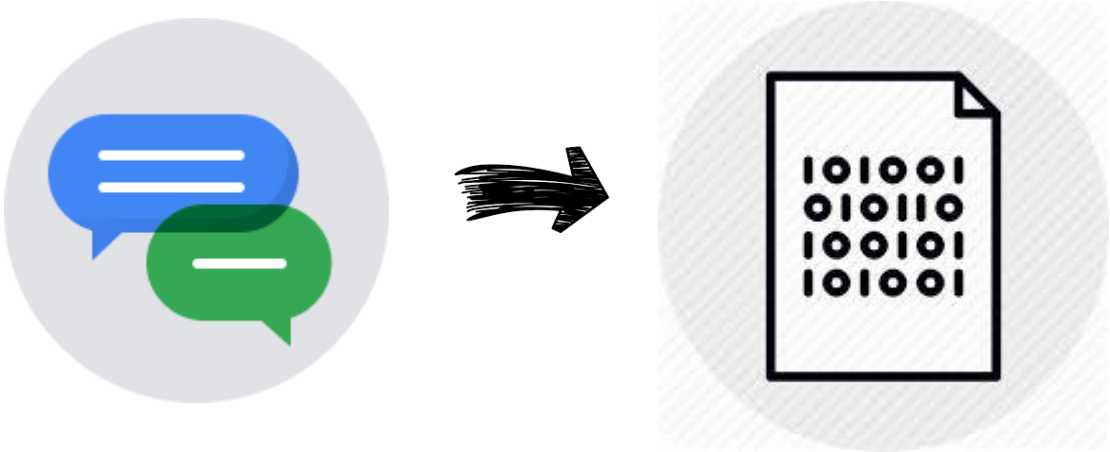
國立清華大學
NATIONAL TSING HUA UNIVERSITY

Huffman Compression

Data Structures Assignment

NTHU EE and CS



https://acm.cs.nthu.edu.tw/problem/12304/

Overview

- Given
  - 1 word
  - N lines, a text

- Task
  - Calculate **character distribution** in the text, do a **stable sort**
  - Build a **Huffman tree**
  - **Encode** the word given at the first line based on the Huffman tree

Input

- The first line of input contains a single positive integer *n* (**number of lines of the text**) and a **word**, separated by a comma, followed by newline
- The next n lines contains zero or more characters(with whitespace), they all end in newlines. This will be the **text.**

Example: Calculate character distribution & sort

have a nice day
love is in the air
have some oranges

Number of h: 3
Number of a: 6
Number of v: 3
...

e	7
a	6
i	4
h	3
v	3
n	3
o	3
s	3
r	2
c	1
d	1

y	1
l	1
t	1
m	1
g	1

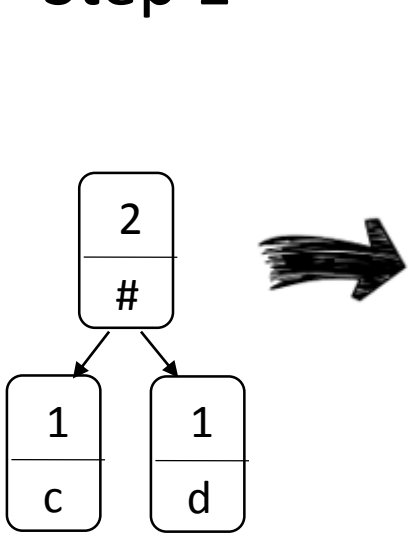
Note: you need to implement an order-stable sorting algorithm. For example, even though "c", "d", "y" … have same char frequency, they are ordered by their appearance order.
This makes **Huffman tree constructing** consist ent.

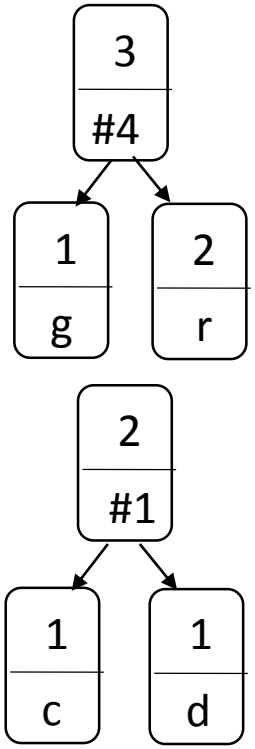
Build Huffman tree with minheap

- The **leaves** contain the characters
- The **inner nodes** are the sum of its leaves
- How to build:
  - Take 2 nodes with **smallest frequency**. Create a node with the **sum of their frequency**.
    - If 2 nodes have the same frequency, choose the **first inputted**
    - If an internal node and a character node have the same frequency, choose the **character node**
  - Add the nodes as the **children** of the new node(smallest frequency as the left child).
  - **Add internal node** to list of internal nodes & **sort**
  - **Remove children** from character distribution
  - Repeat until you only have 1 node in the character distribution (the root)

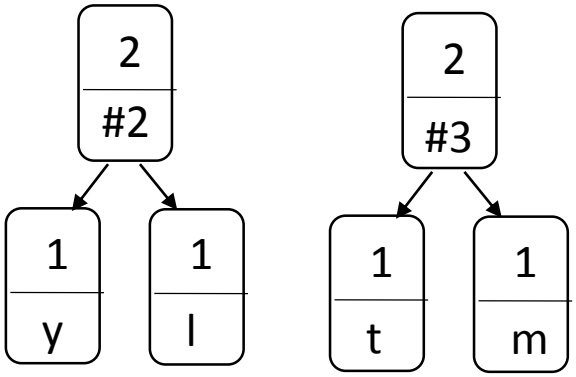
Example: Build Huffman tree with minheap

Step 1





Step 4

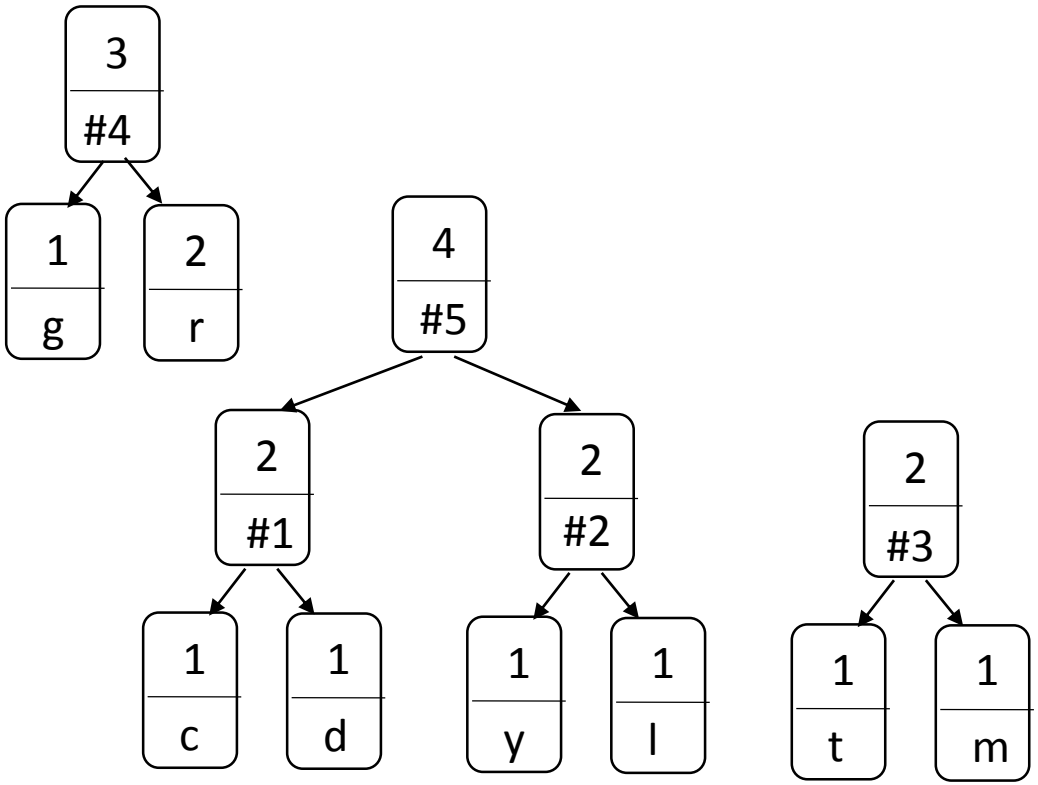


e	7
a	6
i	4
h	3
v	3
n	3
o	3
s	3

#4	3
#1	2
#2	2
#3	2

Example: Build Huffman tree with minheap

Step 6

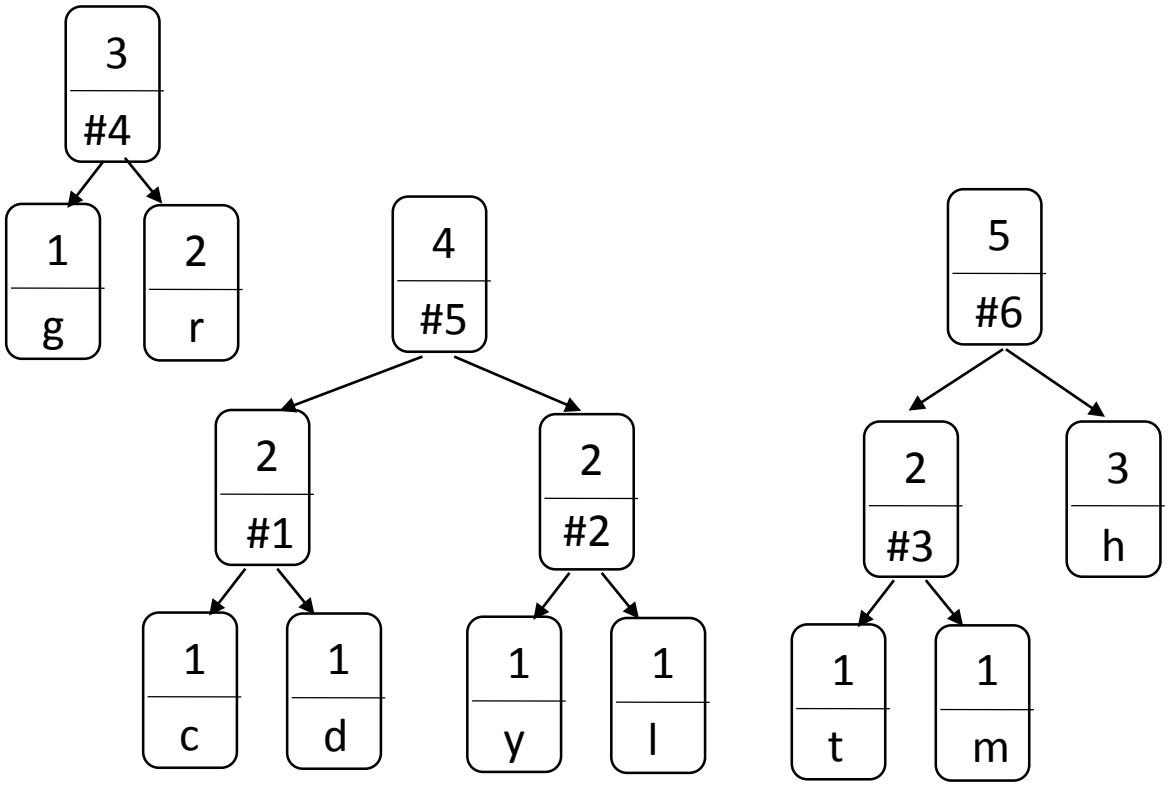


e	7
a	6
i	4
h	3
v	3
n	3
o	3
s	3

#5	4
#4	3
#3	2

Example: Build Huffman tree with minheap

Step 7

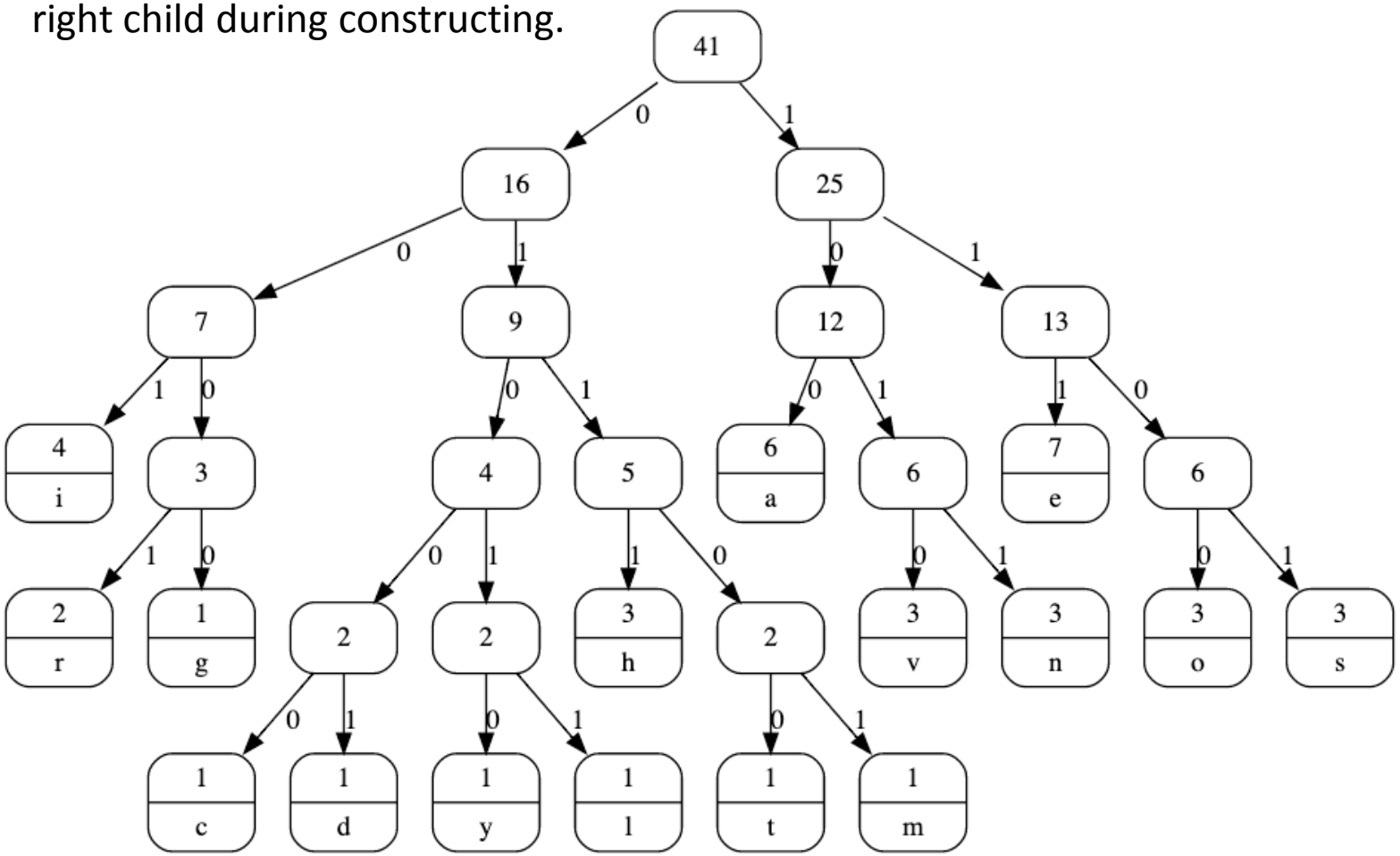


e	7
a	6
i	4
v	3
n	3
o	3
s	3

#6	5
#5	4
#4	3

Example: Build Huffman tree with minheap

Note: There are some nodes swapped because of layout. No matter what layout shows, bit '0' represents following the left child and bit '1' represents following the right child during constructing.

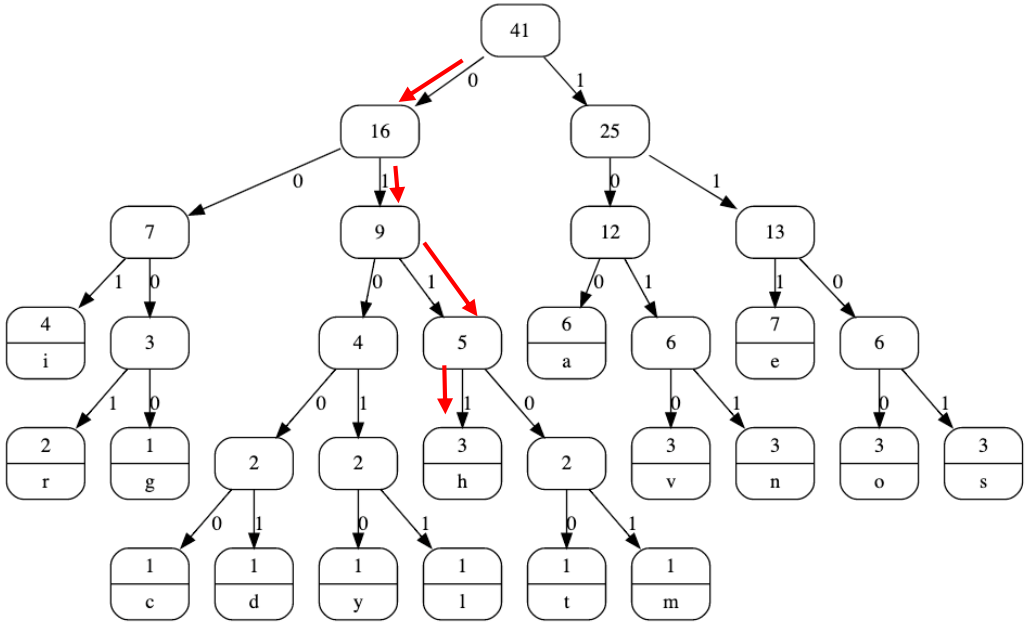


Decode Huffman tree

- The **edge** from a parent to a **right child** is labeled 1.
- The **edge** from a parent to a **left child** is labeled 0.
- The code for a character is its **path** to the node, starting from the root.

Example: Decode Huffman tree

- Letter **h**'s code will be: `0111`



- **hello**'s code will be: `01111110101101011100`

h e l l o

Sample Input

n, and the given word

n lines of text

```
3,hello
have a nice day
love is in the air
have some oranges
```

Sample Output

Encoded word

```
01111110101101011110 0↵
```