Algorithm HW03

Weijun Zhu.

20

1.
```
BinarySearch(search, L):
    start = 0
    end = length(L)-1
    while start <= end:
        middle = ⌊(start+end)/2⌋
        if search < L[middle]:
            end = middle-1
        if search > L[middle]:
            start = middle+1
        if search == L[middle]:
            return middle
```

2.(a). $n_A = P_A \cdot q_A = 97 \times 113 = 10961$

$n_B = P_B \cdot q_B = 127 \times 73 = 9271$

$\phi_A = (P_A - 1)(q_A - 1) = 96 \times 112 = 10752$

$\phi_B = (P_B - 1)(q_B - 1) = 126 \times 72 = 9072$

(b) See code !

$\Rightarrow \begin{cases} K_A = 9721 \\ K_B = 5753 \end{cases}$

(c) See code

(d) See code and conclusion is in annotation

$0 \oplus 0 = 1 \oplus 1 = 0$
$0 \oplus 1 = 1 \oplus 0 = 1$

3. (a). The 1st Pass : $M \oplus K$
The 2nd Pass : $(M \oplus K) \oplus A$
The 3rd Pass : $((M \oplus K) \oplus A) \oplus K = M \oplus A$
Decryption : $(((M \oplus K) \oplus A) \oplus K) \oplus A = M \oplus A \oplus A = M$

(b). The 1st Pass : $M \oplus K_A$
$E_1 = S_{KA}(M)$

$$\begin{array}{r} 11001010 \\ \oplus\ 11001100 \\ \hline 00000110 \end{array}$$

The 2st Pass:
$E_2 = S_{KB}(E_1)$

$$\begin{array}{r} 00000110 \\ \oplus\ 01110101 \\ \hline 01110011 \end{array}$$

The 3rd Pass:
$E_3 = P_{KA}(E_2)$

$$\begin{array}{r} 01110011 \\ \oplus\ 11001100 \\ \hline 10111111 \end{array}$$

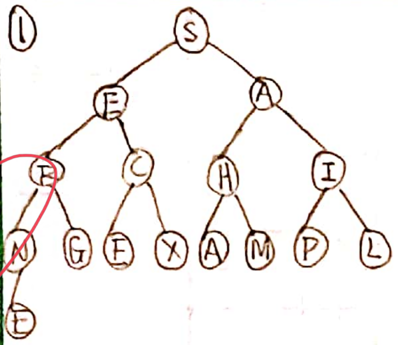
Decryption

$$\begin{array}{r} 10111111 \\ \oplus\ 01110101 \\ \hline 11001010 \end{array}$$

(c). No, there is not a mechanism that Bob receives Alice's message, he knows for sure it is from her.
Three Pass protocol does not offer the method like RSA to verify the Signature. In Three-Pass Protocol, the opponent can pretend to be the receiver to sender or vice-versa. The opponent receives the States from the sender but resends similar but fake states back to the receiver.

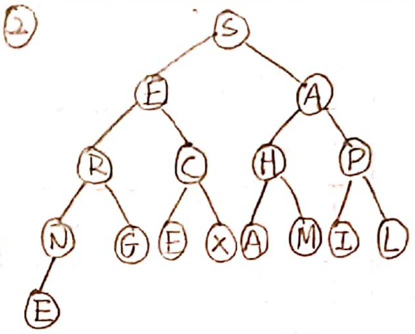
4. ①

20



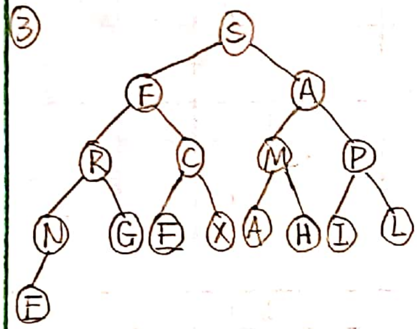
SEARCHINGEXAMPLE

②



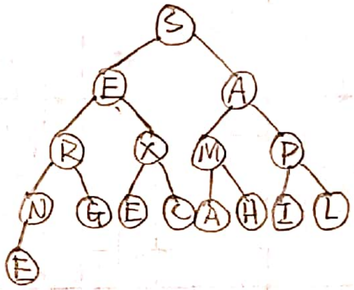
SEARCHPNGEXAMILE

③



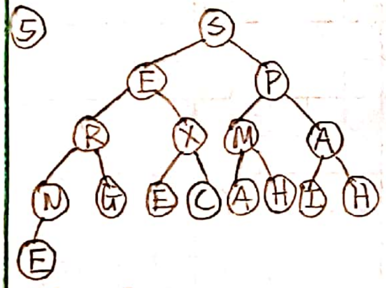
SEARCMPNGEXAHILE

④



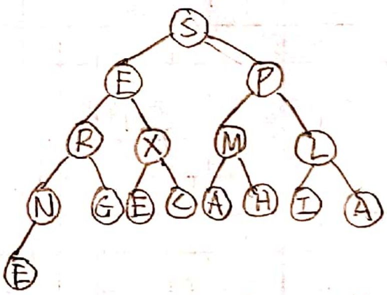
SEARXMPNGECAHILE

⑤



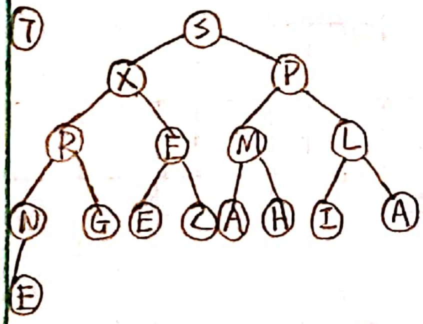
SEPRXMANGECAHIHE

⑥



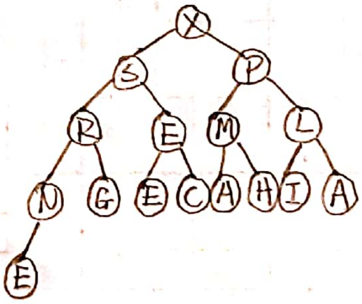
CEPRXMLNGECAHIAE

⑦



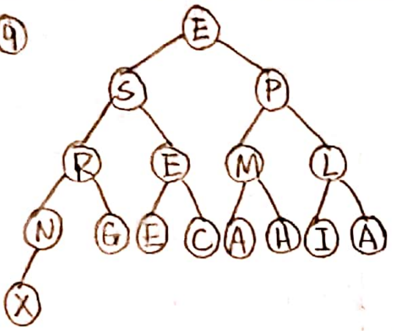
SXPREMLNGECAHIAE

⑧



XSPREMLNGECAHIAE

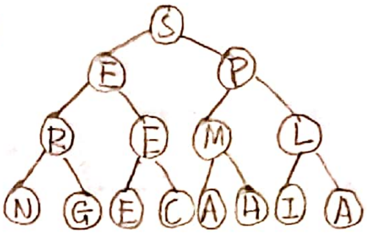
⑨



ESPREMLNGECAHIAX

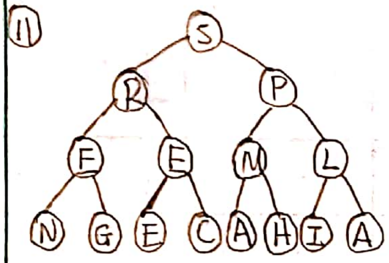
⇝

⑩



SEPREMLNGECAHIAX

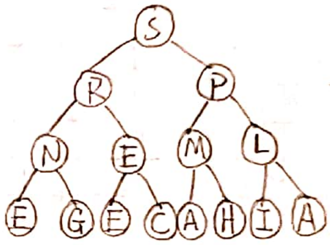
⑪



SRPEEMLNGECAHIAX

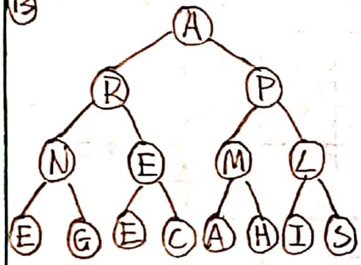
⇝

⑫



SRPNEMLEGECAHIAX

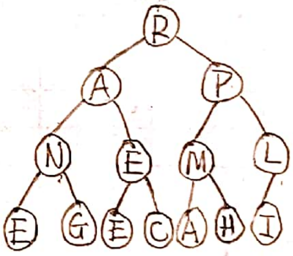
⑬



ARPNEMLEGECAHISX

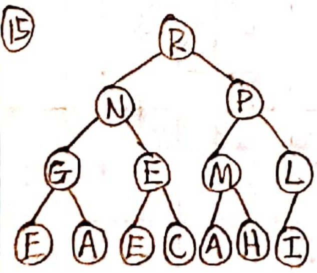
⇝

⑭



PAPNEMLEGECAHISX

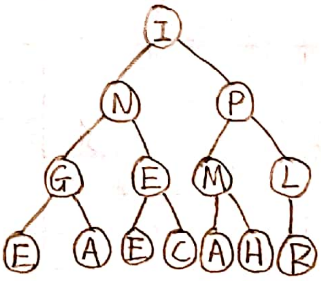
⑮



RNPGEMLEAECAHISX

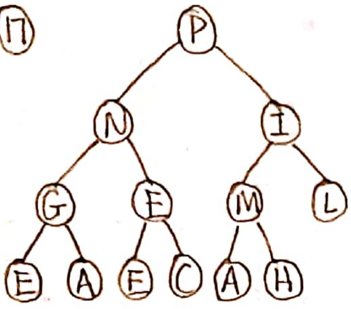
⇝

⑯



INPGEMLEAECAHRSX

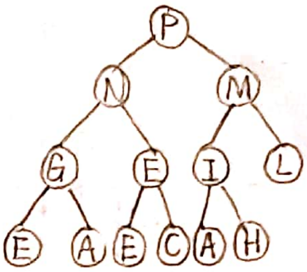
17



PNIGEMLEAECAHRSX

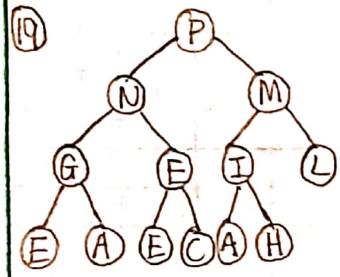
→

18



PNMGEILEAECAHRSX

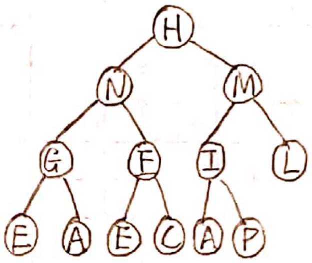
19



PNMGEILEAECAHRSX

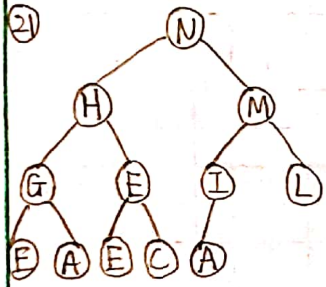
→

20



HNMGEILEAECAPRSX

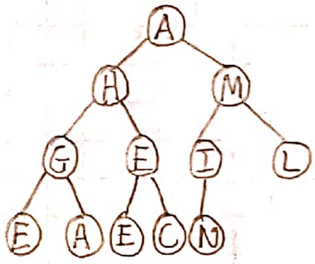
21



NHMGEILEAECAPRSX

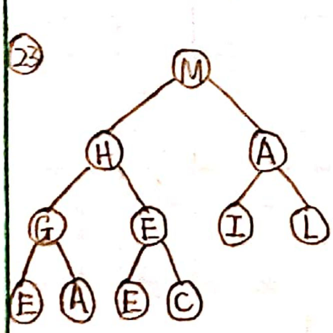
→

22



AHMGEILEAECNPRSX

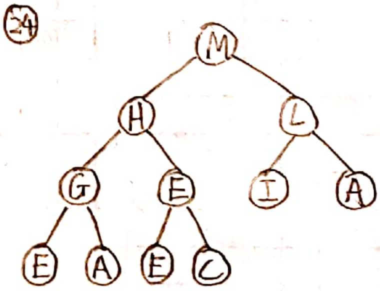
23



MHAGEILEAECNPRSX

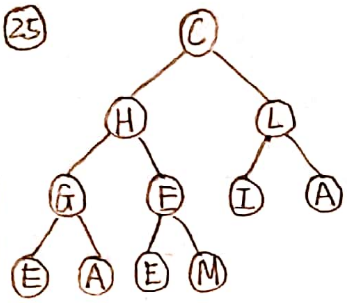
→

24



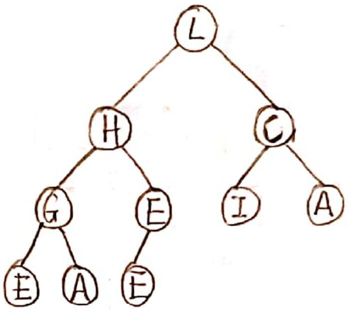
MHLGEIAEAECNPRSX

25



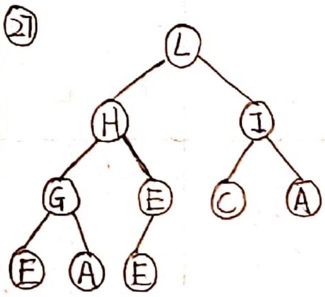
CHLGEIAEAEMNPRSX

26



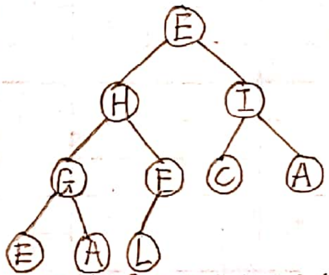
LHCGEIAEAEMNPRSX

27



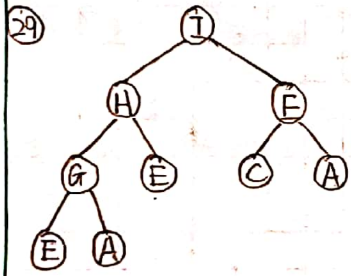
LHIGECAEAEMNPRSX

28



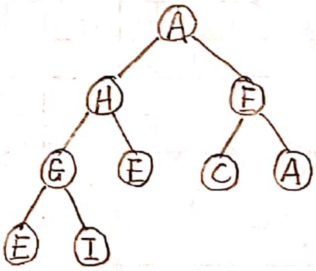
EHIGECAEALMNPRSX

29



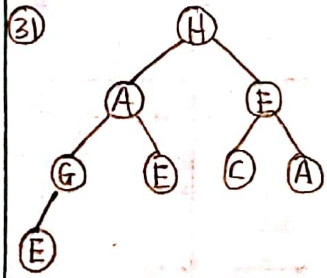
IHEGECAEALMNPRSX

30



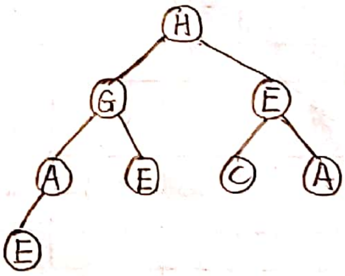
AHEGECAEILMNPRSX

31



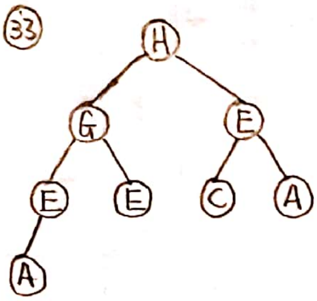
HAEGECAEILMNPRSX

32



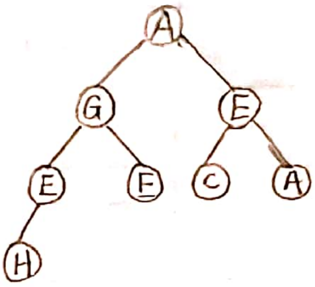
HGEAECAEILMNPRSX

㉝



~→

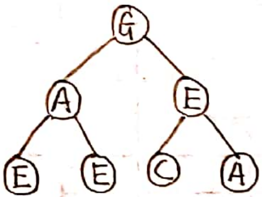
㉞



HGEEECAAILMNPRSX

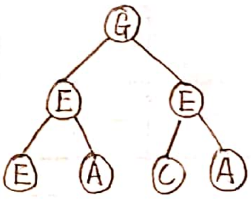
AGEEECAHILMNPRSX

㉟



~→

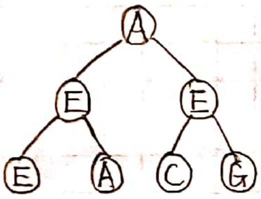
㊱



GAEEECAHILMNPRSX

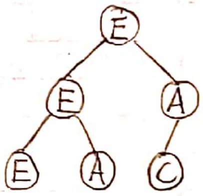
GEEEACAHILMNPRSX

㊲



~→

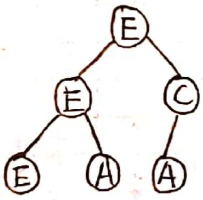
㊳



AEEACGHILMNPRSX

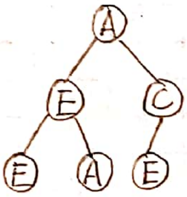
EEAEACGHILMNPRSX

㊴



~→

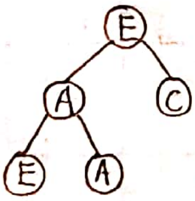
㊵



EECEAAGHILMNPRSX

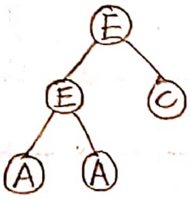
AECEAEGHIMNPRSX

㊶



~→

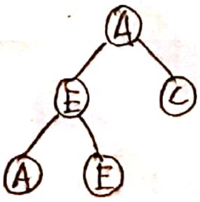
㊷



EACEAEGHILMNPRSX

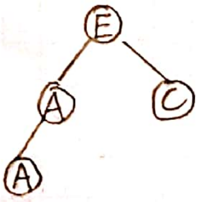
EECAAEGHILMNPRSX

㊸

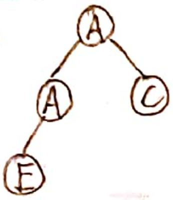


~→

㊹

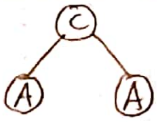


45

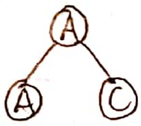


⇝

46



47



AACEEGHILMNPRSX