If we are given with 'n' keys, then there are

$$\boxed{T(n) = \frac{{}^{2n}C_n}{n+1}}$$

possible BST

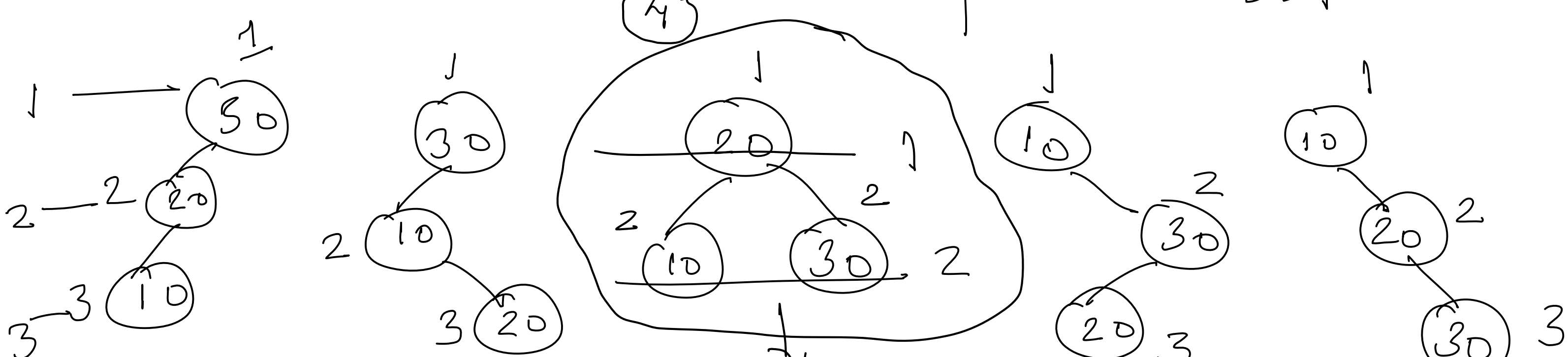
- for e.g $n = 3$ $\boxed{10, 20 \quad 30}$ keys

then $T(n) = \frac{{}^{2\times3}C_3}{3+1}$ possible BST can be made

$$= \frac{{}^{6}C_3}{4} = \frac{6!}{(6-3)!\,3!} = \frac{6\times5\times4\times\cancel{3!}}{3!\ *\ \cancel{3!}}$$

20

$$= \frac{20}{4} = 5 \text{ possible BST}$$



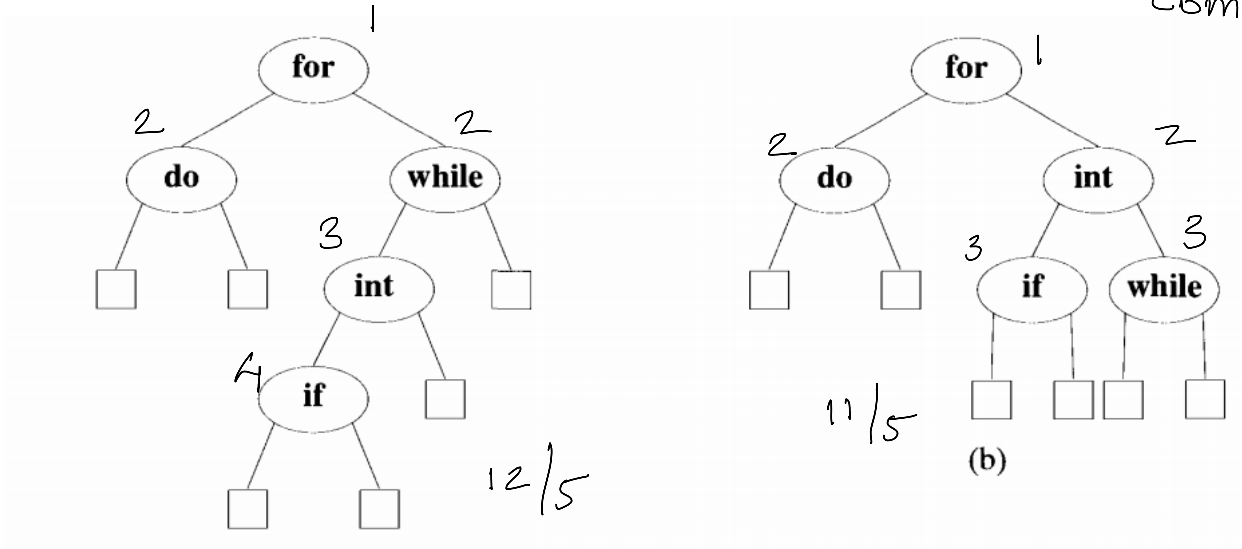
$= 1\times1 + 2\times2 + 3\times3$
$= 14$

$1\times1 + 2\times2 + 2\times2$
$= 9$

- We are given with set of identifiers ⇒ for, do, while, int, if

* **Optimal Binary Search tree problem (OBST)**

- if we see the situation, the number of element Comparisons



- worst case = 4 element Comparison
- worst case = 3 element Comparison

- Avg Comparison = 12/5
- Avg Comparison = 11/5

- We are given with set of identifiers $\{a_1, a_2, a_3, \ldots a_n\}$ s.t $a_1 < a_2 < a_3 < \cdots < a_n$
- Let $p(i)$ be the probability with which we search for a_i (ie probability of success)

- Let $q(i)$ be the probability that the identifier 'x' being searched for is s.t $a_i < x < a_{i+1}$

where $0 \leq i \leq n$

(ie $q(i)$ is the probability of unsuccessful search)

- Therefore $\sum_{1 \leq i \leq n} p(i) + \sum_{0 \leq i \leq n} q(i) = 1$

- To obtain the cost function for BST a fictitious node (external node) is added to leaf nodes

- successful search ends with internal node & unsuccessful search ends with external node

Formulae to remember

* Initially
  1. $w(i,i) = q(i) \rightarrow$ wt function
  2. $c(i,i) = 0 \rightarrow$ cost function
  3. $r(i,i) = 0 \rightarrow$ root function

$$w(i,j) = p(j) + q(j) + w(i, j-1)$$

$$c(i,j) = \min_{i<k\le j} \left\{ c(i,k-1) + c(k,j) \right\} + w(i,j)$$

$r(i,j)$ = is the 'k' value for which $c(i,j)$ gave min value

Problem

Let $n = 4$ $(a_1, a_2, a_3, a_4) = (\text{do}, \text{if}, \text{int}, \text{while})$ → should be in asc order

$$p(1:4) = (\overset{1}{3}, \overset{2}{3}, \overset{3}{1}, \overset{4}{1})$$

$$q(0:4) = (\overset{0}{2}, \overset{1}{3}, \overset{2}{1}, \overset{3}{1}, \overset{4}{1})$$

$\boxed{0 \le i \le 4}$

	0	1	2	3	4
0	$W_{00} = 2$ $C_{00} = 0$ $r_{00} = 0$	$W_{11} = 3$ $C_{11} = 0$ $r_{11} = 0$	$W_{22} = 1$ $C_{22} = 0$ $r_{22} = 0$	$W_{33} = 1$ $C_{33} = 0$ $r_{33} = 0$	$W_{44} = 1$ $C_{44} = 0$ $r_{44} = 0$
1	$W_{01} = 8$ $C_{01} = 8$ $r_{01} = 1$	$W_{12} = 7$ $C_{12} = 7$ $r_{12} =$ 2	$W_{23} = 3$ $C_{23} = 3$ $r_{23} =$ 3	$W_{34} = 3$ $C_{34} = 3$ $r_{34} =$ 4	
2	$W_{02} = 12$ $C_{02} = 19$ $r_{02} = 1$	$W_{13} =$ $C_{13} = 9$ $r_{13} = 12$ 2	$W_{24} = 5$ $C_{24} = 8$ $r_{24} = 3/4$		
3	$W_{03} = 14$ $C_{03} = 25$ $r_{03} = 2$	$W_{14} = 11$ $C_{14} = 19$ $r_{14} = 2$			
4	$W_{04} = 16$ $C_{04} = 32$ $r_{04} = 2$	→	root	node	

$$w(\overset{i}{0},\overset{j}{1}) = p(1) + q(1) + w(0, 1-1)$$
$$= 3 + 3 + w(0,0)$$
$$= 3 + 3 + 2$$

$$\boxed{w(0,1) = 8}$$

$$c(\overset{i}{0},\overset{j}{1}) = \min_{0<k\leq 1} \{c(0,0) + c(1,1)\} + w(0,1)$$

$k = 1$

$$= \min\{0 + 0\} + w(0,1)$$

$$\boxed{c(0,1) = 8}$$

$$\boxed{r(0,1) = 1}$$

$$w(0,2) = p(2) + q(2) + w(0,1)$$
$$= 3 + 1 + 8$$
$$\boxed{w(0,2) = 12}$$

?

$$c(0,2) = \min_{0 < k \le 2} \left\{ \overbrace{c(0,0) + c(1,2)}^{k=1},\ \underbrace{c(0,1) + c(2,2)}_{k=2} \right\} + w(0,2)$$

$k = 1, 2$

$$= \min \{ 0 + 7,\ 8 + 0 \} + 12$$

$$\boxed{c(0,2) = 19} \longrightarrow \boxed{r(0,2) = 1}$$

$$c(1,2) = \min_{1 < k \le 2} \{ c(1,1) + c(2,2) \} + w(1,2)$$

$k = 2$

$$\boxed{c(1,2) = 7}$$

?

$$w(1,2) = p(2) + q(2) + w(1,1)$$
$$= 3 + 1 + q(1)$$
$$= 3 + 1 + 3$$
$$\boxed{w(1,2) = 7}$$

$$w(0,3) = p(3) + q(3) + w(0,2)$$
$$= 1 + 1 + 12$$
$$\boxed{w(0,3) = 14}$$

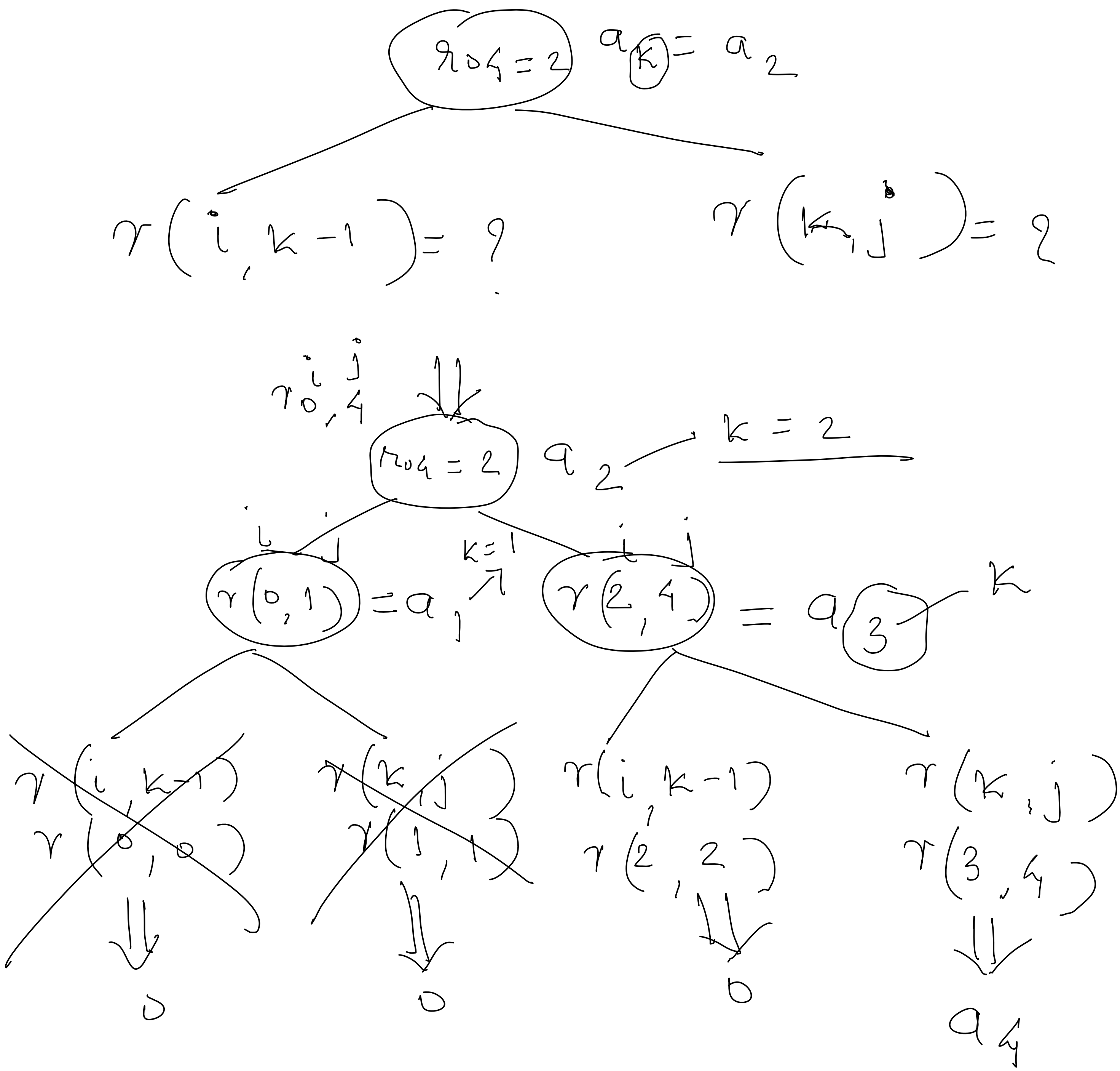
?

$$c(0,3) = \min_{0 < K \leq 3} \left\{ \begin{array}{l} c(0,0) + c(1,3), \\ c(0,1) + c(2,3), \\ c(0,2) + c(3,3) \end{array} \right\} + w(0,3)$$

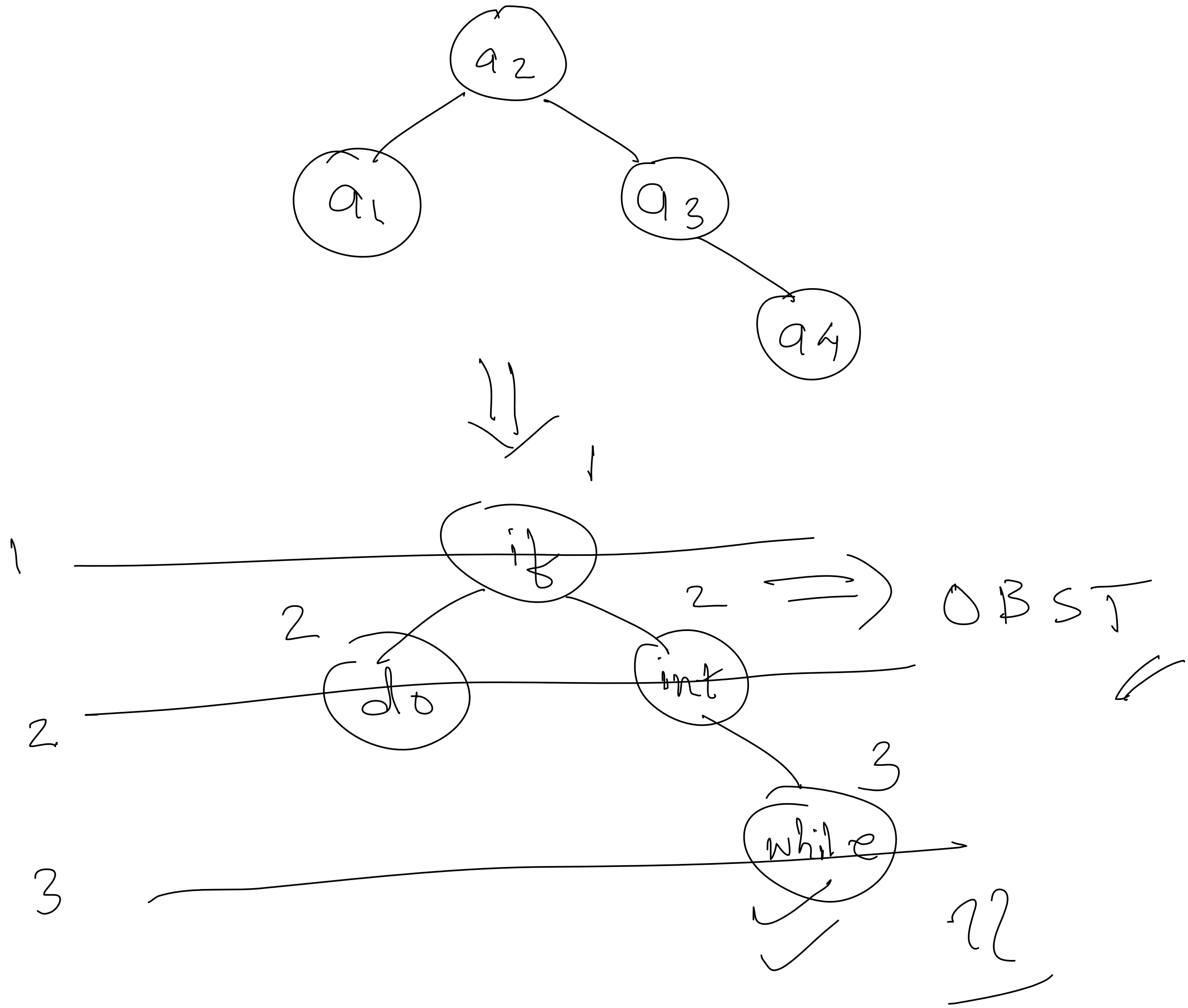
$$\Downarrow$$

$$K = 1, 2, 3$$

?



r04 = 2
a k = a 2
r (i , k - 1) = ?
r (k , j) = ?
i j
r 0 , 4
r04 = 2
a 2
k = 2
i j
k = 1
i j
r (0 , 1) = a 1
r (2 , 4) = a 3
k
r (i , k - 1)
r (0 , 0)
r (k , j)
r (1 , 1)
r (i , k - 1)
r (2 , 2)
r (k , j)
r (3 , 4)
0
0
0
a 4



a2
a1
a3
a4
1
if
2
2
do
int
2
3
while
3
OBST